
BYTE
NOVEMBER 1977
the small systems journal
VOLUME 2, Number 11
$2.00 in USA
BRUCE HOLLOWAY

You can now have the industry's finest microcomputer with that all-important disk drive

The professional-grade microcomputer for professionals

Shown with optional bench cabinet

YOU CAN GET THAT ALL-IMPORTANT SOFTWARE, TOO

Loading your programs and files will take you only a few seconds with the new Cromemco Z-2D computer.

You can load fast because the Z-2D comes equipped with a 5″ floppy disk drive and controller. Each diskette will store up to 92 kilobytes.

Diskettes will also store your programs inexpensively—much more so than with ROMs. And ever so much more conveniently than with cassettes or paper tape.

The Z-2D itself is our fast, rugged, professional-grade Z-2 computer equipped with disk drive and controller. You can get the Z-2D with either single or dual drives (dual shown in photo).

CROMEMCO HAS THE SOFTWARE

You can rely on this: Cromemco is committed to supplying quality software support.

For example, here's what's now available for our Z-2D users:

CROMEMCO FORTRAN IV COMPILER: a well-developed and powerful FORTRAN that's ideal for scientific use. Produces optimized, relocatable Z-80 object code.

CROMEMCO 16K DISK BASIC: a powerful pre-compiling interpreter with 14-digit precision and powerful I/O handling capabilities. Particularly suited to business applications.

CROMEMCO Z-80 ASSEMBLER: a macro-assembler that produces relocatable object code. Uses standard Z-80 mnemonics.



ADVANCED CONTROLLER CARD

The new Z-2D is a professional system that gives you professional performance.

In the Z-2D you get our well-known 4-MHz CPU card, the proven Z-2 chassis with 21-slot motherboard and 30-amp power supply that can handle 21 cards and dual floppy drives with ease.

Then there's our new disk controller card with special features:

- Capability to handle up to 4 disk drives
- A disk bootstrap Monitor in a 1K 2708 PROM
- An RS-232 serial interface for interfacing your CRT terminal or teletype
- LSI disk controller circuitry

Z-2 USERS:
Your Z-2 was designed with the future in mind. It can be easily retrofitted with everything needed to convert to a Z-2D. Only $935 kit; or $1135 for assembled retrofit package.

We're able to put all of this including a UART for the CRT interface on just one card because we've taken the forward step of using LSI controller circuitry.

STORE/FACTORY

Contact your computer store or Cromemco factory now about the Z-2D. It's a real workhorse that you can put to professional or OEM use now.

Kit: Z-2D with 1 disk drive (Model Z2D-K) .... $1495.
Assembled: Z-2D fully assembled and tested (Model Z2D-W) .... $2095.
Additional disk drive (Model Z2D-FDD) .... $495.

SOFTWARE

(On standard IBM-format soft-sectored mini diskettes)

16K BASIC (Model FDB-S) $95
FORTRAN IV (Model FDF-S) $95
Z-80 Assembler (Model FDA-S) $95



Cromemco
incorporated
Specialists in computers and peripherals
2400 CHARLESTON RD., MOUNTAIN VIEW, CA 94043 • (415) 964-7400

Circle 40 on inquiry card.

TV DAZZLER

4K RAM

8K BYTESAVER

16K RAM

16K PROM CARD

CPU CARD

ANALOG I/O

JOYSTICK CONSOLE

TU-ART

To make your computer more useful– a wide choice of memory, I/O, CPU

Your computer's usefulness depends on the capability of its CPU, memories, and I/O interfaces, right?

So here's a broad line of truly useful computer products that lets you do interesting things with your Cromemco Z-1 and Z-2 computers. And with your S-100-compatible Altairs and IMSAIs, too.

CPU

- Z-80 MICROPROCESSOR CARD. The most advanced μP card available. Forms the heart of our Z-1 and Z-2 systems. Also a direct replacement for Altair/IMSAI CPUs. Has 4-MHz clock rate and the power of the Z-80 μP chip. Kit (Model ZPU-K): $295. Assembled (Model ZPU-W): $395.

MEMORIES

- 16K RAM. The fastest available. Also has bank-select feature. Kit (Model 16KZ-K): $495. Assembled (Model 16KZ-W): $795.
- 4K RAM. Bank-select allows expansion to 8 banks of 64K bytes each. Kit (Model 4KZ-K): $195. Assembled (Model 4KZ-W): $295.
- THE BYTESAVER — an 8K capacity PROM card *with integral programmer*. Uses high-speed 2708 erasable PROMs. A must for all computers. Will load 8K BASIC into RAM in less than a second. Kit (Model BSK-0): $145. Assembled (Model BSW-0): $245.
- 16K CAPACITY PROM CARD. Capacity for up to 16K of high-speed 2708 erasable PROM. Kit (Model 16KPR-K): $145. Assembled (Model 16KPR-W): $245.

I/O INTERFACES

- FAST 7-CHANNEL DIGITAL-ANALOG I/O. Extremely useful board with 7 A/D channels and 7 D/A channels. Also one 8-bit parallel I/O channel. Kit (Model D + 7A-K): $145. Assembled (Model D + 7A-W): $245.
- TV DAZZLER. Color graphics interface. Lets you use color TV as full-color graphics terminal. Kit (Model CGI-K): $215. Assembled (Model CGI-W): $350.
- DIGITAL INTERFACE (OUR NEW TU-ART). Interfaces with teletype, CRT terminals, line printers, etc. Has not one but *two serial I/O ports* and *two 8-bit parallel I/O ports* as well as 10 on-board interval timers. Kit (Model TRT-K): $195. Assembled (Model TRT-W): $295.
- JOYSTICK. A console that lets you input physical position data with above Model D + 7 A/D card. For games, process control, etc. Contains speaker for sound effects. Kit (Model JS-1-K): $65. Assembled (Model JS-1-W): $95.

PROFESSIONAL QUALITY

You get first-class quality with Cromemco.

Here are actual quotes from articles by independent experts: "The Cromemco boards are absolutely beautiful" . . . "The BYTESAVER is tremendous" . . . "Construction of Cromemco I/O and joystick are outstanding" . . . "Cromemco peripherals ran with no trouble whatsoever."

Everyone agrees. Cromemco is tops.

STORES/MAIL

So count on Cromemco. Look into these Cromemco products at your store. Or order by mail from the factory.

We wish you pleasure and success with your computer.



Cromemco
i n c o r p o r a t e d
Specialists in computers and peripherals
2432 CHARLESTON RD., MOUNTAIN VIEW, CA 94043 • (415) 964-7400

Circle 40 on inquiry card.

In the Queue

BYTE NOVEMBER 1977
Volume 2 Number 11

Foreground

Background

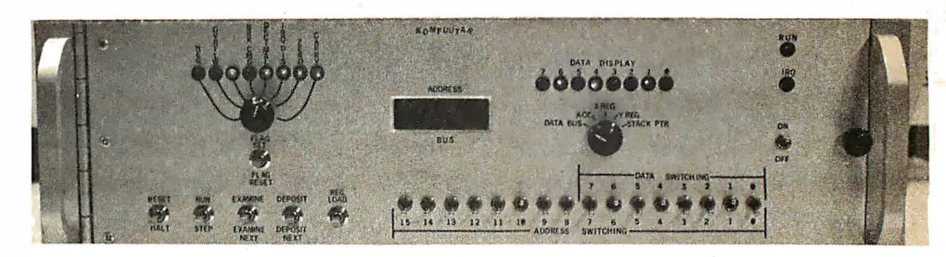
Nucleus

Cover by Bruce Holloway

BYTE is published monthly by BYTE Publications Inc, 70 Main St, Peterborough NH 03458. Address all mail except subscriptions to above address; phone (603) 924-7217. Address all editorial correspondence to the editor at the above address. Unacceptable manuscripts will be returned if accompanied by sufficient first class postage. Not responsible for lost manuscripts or photos. Opinions expressed by the authors are not necessarily those of BYTE. Address all subscriptions, change of address, Form 3579, and fulfillment complaints to BYTE Subscriptions, PO Box 361, Arlington MA 02174; phone (617) 646-4329. Second class postage paid at Peterborough NH 03458 and at additional mailing offices—USPS Publication No. 102410. Subscriptions are $12 for one year, $22 for two years, and $32 for three years in the USA and its possessions. Add $5.50 per year for subscriptions to Canada and Mexico. $25 for a one year subscription by surface mail worldwide. Air delivery to selected areas at additional rates available upon request. $25 for a one year subscription by air delivery to Europe. Single copy price is $2.00 in the USA and its possessions, $2.40 in Canada and Mexico, and $3.50 in Europe, and $4.00 elsewhere. Foreign subscriptions and sales should be remitted in United States funds. Printed in United States of America.

In This BYTE

page 94



In this issue, author Steve Ciarcia begins what we expect to become a regular feature in BYTE: Ciarcia's Circuit Cellar. Steve, a senior engineering consultant to the aerospace industry by profession, is a rare combination of writer and tinkerer. The conceptual model he brings to his interactive column format is that of the late C L Stong's stewardship of "The Amateur Scientist" in *Scientific American*, but with an emphasis on hardware and software combinations to accomplish interesting applications of personal computing systems. Steve welcomes feedback from readers. . .CH

Games and models which employ moving objects require some attention to details of motion as simulated by a computer program. Beginning a series of articles on the subject of moving objects, Stephen P Smith's Simulation of Motion: An Improved Lunar Lander Algorithm shows how a real time game can incorporate models of motion in more than one dimension.

Donald T Piele shows that a computer fair doesn't have to be big to be good. A Minicomputer Fair: Tiny and Personal describes the University of Wisconsin's efforts to produce their own micro extravaganza, which drew over 700 attendees. Readers may get some ideas about putting on shows of their own based on Professor Piele's experiences.

What might not be appreciated by the neophyte is the fact that an interrupt driven clock suggests other uses besides keeping time. In M F Smith's article on Using Interrupts for Real Time Clocks you'll find a simple timekeeping algorithm, and a sketch of how it can be extended to share processor time between two different processes.

Do you occasionally find incorrect data in your computer when you know you entered the correct information and processed it with a reliable program? Does your computer do strange things every time the washing machine or furnace turns on? Perhaps your problem is voltage transients. John McCain writes about Spikes: Pesky Voltage Transients and How to Minimize Their Effects.

If you want to post a calendar of events in your computer's memory with a resolution of 1 second, a mere three integrated circuits added to an existing LSI digital clock can turn it into a source of time information for your computer. Use Robert Grappel's article in this issue to find an answer to the metaphorical question: "Does Anybody Know What Time It Is?"

Any regular source of interrupts can be used as the key element in a simple real time clock for the typical personal computer. James R Sneed shows how to create such an interrupt source, then program a 6502 to generate internal variables for hours, minutes, seconds and 1/15th seconds of the day in his article on Adding an Interrupt Driven Real Time Clock.

If you do a lot of mathematical calculations on your microcomputer, you'll enjoy reading Floating Point Arithmetic by Burt Hashizume. Find out how to add an economical floating point package to your system and improve your number crunching facilities.

An excellent way to learn about computers is to build one yourself. Hilary D Jones shows that this is not such a terrifying task. Read Building a Computer From Scratch and find out how to construct a working (albeit limited) computer for under $70 (plus the price of a power supply).

Occasionally readers ask for detail plans of computer systems. David Brader, a BYTE reader from Electric City WA, has implemented an excellent piece of homebrew craftsmanship in his Kompuutar system based on the MOS Technology 6502 processor. In this issue, we provide David's complete design for the central processor, control panel interface, and serial terminal interface of a general purpose computer.

Frequency counters are useful tools for a variety of applications. Perry Lynne shows you how to add one to your microcomputer in Implementing an LSI Frequency Counter. His design takes advantage of the Intel 8253 programmable interval timer (as well as the power of the microprocessor) to produce a design that is both accurate and economical.

How do you make an 8 bit machine emulate a more comprehensive design? In his article, SWEET16: The 6502 Dream Machine, Stephen Wozniak details the design and functions of a low level interpreter for 16 bit operations which extend the functions of the more limited 8 bit 6502 processor.

Continuing the theme of real time and how to keep track of it, G A R Trollope provides an example of the interrupt driven approach, implemented through the IRQ interrupt line of a 6800 processor with a PIA port. Do You Need Real Time? If so, turn to this article.

The game of NIM is well-known in the annals of computer lore, but many people have had no contact with it. Irwin Doliner presents us with a version of the game and supplies us with the design theory behind it in his article, NIMBLE: The Ultimate NIM?

Verbatim™

The whole message is quality.

You want to record your message verbatim—word for word—whether it's bits, bytes or "Dear Folks" translated into word processor language.

Our objective in manufacturing recording media for the electronics industry—digital tape cassettes, floppy disks, mag cards, computer cartridges—is to give you the finest, the best, the most dependable, the most cost-effective.

That means rugged, long-lived, abrasion-resistant recording media with superior magnetic qualities. If we made tires, they'd be steel-belted radials.

We delivered our first digital grade certified tape cassettes back in the beginning, 1969. We made the first commercial 3740-compatible floppy disks that didn't bear IBM's name. And the first Flippy® reversible flexible disks with *anyone's* name on them. The first mini data cassette is ours. And we've got the newest miniature flexible disk, the MD 525.

Now, **Verbatim** media. It's a new formulation of ferric oxides, an advanced macromolecular binder system to adhere it to the tough polyester film, and a process control system that demands over 200 separate quality checks before the material is cut, packaged, and certified to be 100% error-free.

The final quality check? "Make it pretty!" Our production people tell us that magnetic recording media is one of the rare instances in manufacturing where aesthetic appearance translates directly into final product quality. It has to look beautiful to work beautifully.

We have the formulas, the machines, the technology to make high quality recording media. But it takes the best people in the industry to deliver Verbatim disks, cards, cartridges and cassettes. You'll find them at your favorite retail computer store.



Circle 68 on inquiry card.

Information Terminals Corp.®
Sunnyvale, California



The Compleat Robotics Experimenter

By Carl Helmers

On August 16 1977 I received one of those refreshing and intoxicating articles (or rather group of articles) which makes the combined intellectual and emotional joys of creating a magazine once a month rise to new heights. This group of articles is a basic background tutorial on biological inputs to the field of robotics and artificial intelligence, written for the personal computing experimenter by Ernest W Kent, a professor in the department of psychology of the University of Illinois at Chicago Circle. It is one of those articles, like Ralph Hollis' article on NEWT in the June 1977 BYTE, which gets instant high priority due to the subject matter and style of presentation. (Readers should see the beginning of the series in early 1978.)

I call the twin subjects of robotics and artificial intelligence "hot" ideas for BYTE based on reader interest as expressed in the BOMB poll's responses to Ralph Hollis' article on NEWT and Mike Wimble's articles (among others) on various artificial intelligence concepts. Inspired by receipt of Dr Kent's articles, the theme of this editorial is the concept of smart machines and related robotic mechanisms as a fertile field for experimentation with design and implementation. What are the categories and classes of experimentation which are relevant to artificial intelligence and robot design? Why are we (experimenters all) so fascinated by the simulation of life? What are the topics of study needed to become "the complete robotics experimenter?" What will we see over the course of the next decade or so, as personal computers become the refined personal software development systems needed to support private robotics research?

It often helps to draw inspiration from fiction, an element of our culture which has been present from its beginnings in the allegorical tales of primitive religions to the sophisticated and future oriented technological fiction tales of contemporary film, television and printed media. Fictional representations of plots, scenarios and tales are a sort of logical game practiced by creators, logical games with very real emotional and value orientations which stimulate thought about real problems while providing an interesting and enjoyable diversion for users of the art. Technological fiction, of which science fiction is a proper subset, is the appropriate contemporary place to turn for inspiration regarding the very comtemporary possibility of ingenious and useful automatons guided by artificial intelligence.

A particular science fiction tale which has been one of my greatest emotional inputs regarding the positive values of technology in human culture is a tale entitled *Door Into Summer*, by Robert Heinlein. First published in the 1950s, this now outdated tale of the near future (1970 is the year when the action commences with flashbacks to the fictional 1960s) is perhaps the one science fiction story which maps most closely to the current technological milieu of the smart machines made possible by microprocessor technology. Anyone who is seriously interested in practical use of robotic technology and smart machines should read this book as a source of background information and ideas about what is or might be possible. (The actual plot is a well constructed romantic tale in spite of its use of that logical trap which is the time travel *deus ex machina*.)

The inspiration to be drawn from the story of *Door Into Summer* is that of an exciting time when technology has advanced to the threshold of intelligent robotic mechanisms mass-produced for use in mundane tasks. It is the era of Drafting Dan (automated intelligent drafting machine), Hired Girl (automated housekeeping robot), and numerous similar specialized devices. Some of these fictional concepts have already been implemented in practice, especially in the area of automated aids to the production of capital goods. The idea of Drafting Dan, the intelligent drafting device, is actually in use on a small scale today but with a far higher degree of refinement and intelligence: I refer to the various computer aided design techniques utilizing graphic displays and computational support in fields as diverse as airplane design, computer design, and architecture. Others among the concepts in Robert Heinlein's story have yet to be implemented with any degree of perfection or widespread use.

The parallels between *Door Into Summer* and the current era are many. In the fictional account, technology has developed

Continued on page 138

The complete $655 line printer.

It's ready to plug in, has an 80-column format, a remarkable MTBF and is 14 times faster than a teletype!



Breaking the hardcopy barrier

It's finally happened! The Axiom EX-800 provides full performance hardcopy at a price compatible with today's low cost micros. This little 80-column machine zips along at 160 characters per second (14 times faster than a teletype)—at a breakthrough single quantity price of $655 for a complete printer.

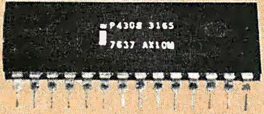
When we say complete we mean it

The EX-800 is a stand-alone unit with case, power supply, 96 character ASCII generator and interface, paper roll holder, infra-red low paper detector, bell, and multi-line asynchronous input buffer. You won't find these standard features on any other printer, regardless of price!

Our only option

Our printer is so complete, that we offer only one option. A serial interface (RS 232C or current loop) good for 16 baud rates from 50 to 19,200 and thoughtfully provided with a switch for either Centronics or Tally compatibility. Might we call it a Tally-whacker? At $85.00 it certainly should be!

Built-in LSI microprocessor



The heart of the EX-800 is a printed circuit card, containing a custom LSI chip made by Intel to Axiom specifications, which controls all printer functions. Microprocessor power means flexibility. Such as the built-in self test routine and variable character size. It also means reliability. Several industry surveys have shown LSI to be many times more reliable than equivalent conventional circuitry.

THIS LIFE-SIZE SAMPLE SHOWS THE 80-COLUMN PRINTOUT FROM AXIOM'S EX-800 PRINTER

There are 3 character sizes (upper and lower case) which can be MIXED.

This can have the same effect as UNDERLINING or changing COLOR.

The advantages of electrosensitive printing

The EX-800 can print 80, 40, or 20 characters across the five inch wide electrosensitive paper. Under software control, single characters or words may be printed larger for emphasis. The permanence of the hardcopy is archival, because once the aluminum coating has been removed, there is no way to put it back. It's unaffected by sunlight, moisture or heat. Although the printer doesn't provide multiple copies, excellent quality photocopies can be made from the high contrast printout. Also, the paper is inexpensive and readily available, costing about 1¢ for an 8½ x 11" equivalent.

Light, small, quiet, reliable, and versatile

Our EX-800 weighs in at 12 pounds, is just 9½ inches wide, 4 inches high, and 11 inches deep, and is delightfully quiet which makes it ideal for office and other low noise environments. The simple print mechanism is virtually maintenance free. In fact, tests show an incredible MTBF, many times greater than impact printers. This versatile printer is the ideal mate for micros, minis, CRTs, instruments and systems.

Just unbox and plug it in

That's all you have to do to the Axiom EX-800 — apart from pay for it, and at $655 that's almost a pleasure.

Send to: AXIOM
5932 San Fernando Rd.,
Glendale, CA 91202

- ☐ Urgent. Please phone me at ext ____
- ☐ Have rep contact me
- ☐ I'd like to have a demonstration
- ☐ Send lit including sample of printout

Name ____
Company ____
Dept ____
Address ____
City ____ State ____
Zip ____ Telephone ____

AXIOM
AXIOM CORPORATION
5932 San Fernando Rd., Glendale, CA 91202 • (213) 245-9244 • TWX 910-497-2283

Circle 8 on inquiry card.

Check out TI's new 4K static RAMs.

They've got everything you ever liked about the 2102.

And more.

	2102 1K Static RAM	TI's New 4K Static RAMs	
SIMPLE TO USE	✓	✓	Like the popular 2102, TI's new 4K static RAMs are easy to use. Minimize system overhead; no refresh; simple addressing. It's easy!
NO CLOCKS. NO TIMING STROBES. ADDRESSES VALID ETC... ACCESS TIME DATA OUT VALID	✓	✓	No clocking needed for TI's fully static 4K RAMs. No edges. Just present an address to the selected device and data can be read at access time. That's it.

Circle 152 on inquiry card.

	2102 1K Static RAM	TI's New 4K Static RAMs	
FULLY STATIC. ACCESS TIME = CYCLE TIME.	✓	✓	Fully static RAMs are totally asynchronous. Require no precharge or recovery time. Access and cycle times are always the same.
DATA VALID – NO TIME LIMIT.	✓	✓	Fully static RAMs offer output data that are valid as long as the address is valid. Makes designing straightforward. No limit on output valid time. No extra circuitry.
SINGLE +5 V SUPPLY. FULLY TTL COMPATIBLE.	✓	✓	Just one +5 V supply needed. Full TTL compatibility on all inputs and outputs with full 400 mV guaranteed dc noise immunity.
±10% TOLERANCE SUPPLY.		✓	Improved power supply tolerance means less stringent regulation. Less cost.
HIGH SPEED.		✓	TI's new 4K static RAMs take up where the 2102 left off. Offering a wide choice of speeds from 150 ns to 450 ns maximum access/minimum cycle. Plenty of performance to match today's and tomorrow's CPUs.
LOW POWER.		✓	Compare the power savings of the new 4K statics to the low power 21L02. For equivalent speed, the new TMS 4044 uses 63% less power. For super low standby-power/battery-backup operation, use the pin-compatible 20-pin TMS 4046/47 Series. Data is retained down to 10 mW.
HIGH DENSITY 18-PIN PACKAGE.		✓	The new 4K statics come in industry-standard, 18-pin packages, plastic or ceramic. A density improvement of almost four-to-one over 2102s.

DATA VALID – NO TIME LIMIT.

ADDRESSES
VALID
ACCESS TIME
DATA OUT
VALID
NO TIME LIMIT

HIGH SPEED.

2102
TI 4K STATICS
1000 900 800 700 600 500 400 300 200 100
ACCESS/CYCLE TIME (ns)

Max. Access	4K x 1	1K x 4
150 ns	TMS 4044-15	TMS 4045-15
200 ns	TMS 4044-20	TMS 4045-20
250 ns	TMS 4044-25	TMS 4045-25
300 ns	TMS 4044-30	TMS 4045-30
450 ns	TMS 4044-45	TMS 4045-45

LOW POWER.

Parameter	Four Low Power 1K Static RAMs (2102AL-2)	One TI 4K Static RAM (TMS 4044-25)
Max. Access Min. Cycle	250 ns	250 ns
Operating Power (Max.)	1368 mW	500 mW
Package(s)	4 x 16 pin	1 x 18 pin
Board Area Ratio	3.7	1.0

HIGH DENSITY 18-PIN PACKAGE.

4044 4045

FOR DATA SHEETS, application reports, price and delivery, call your nearest authorized TI distributor or TI field sales office.

TEXAS INSTRUMENTS
INCORPORATED

93192

Circle 152 on inquiry card.

Ciarcia's Circuit Cellar

Memory Mapped IO

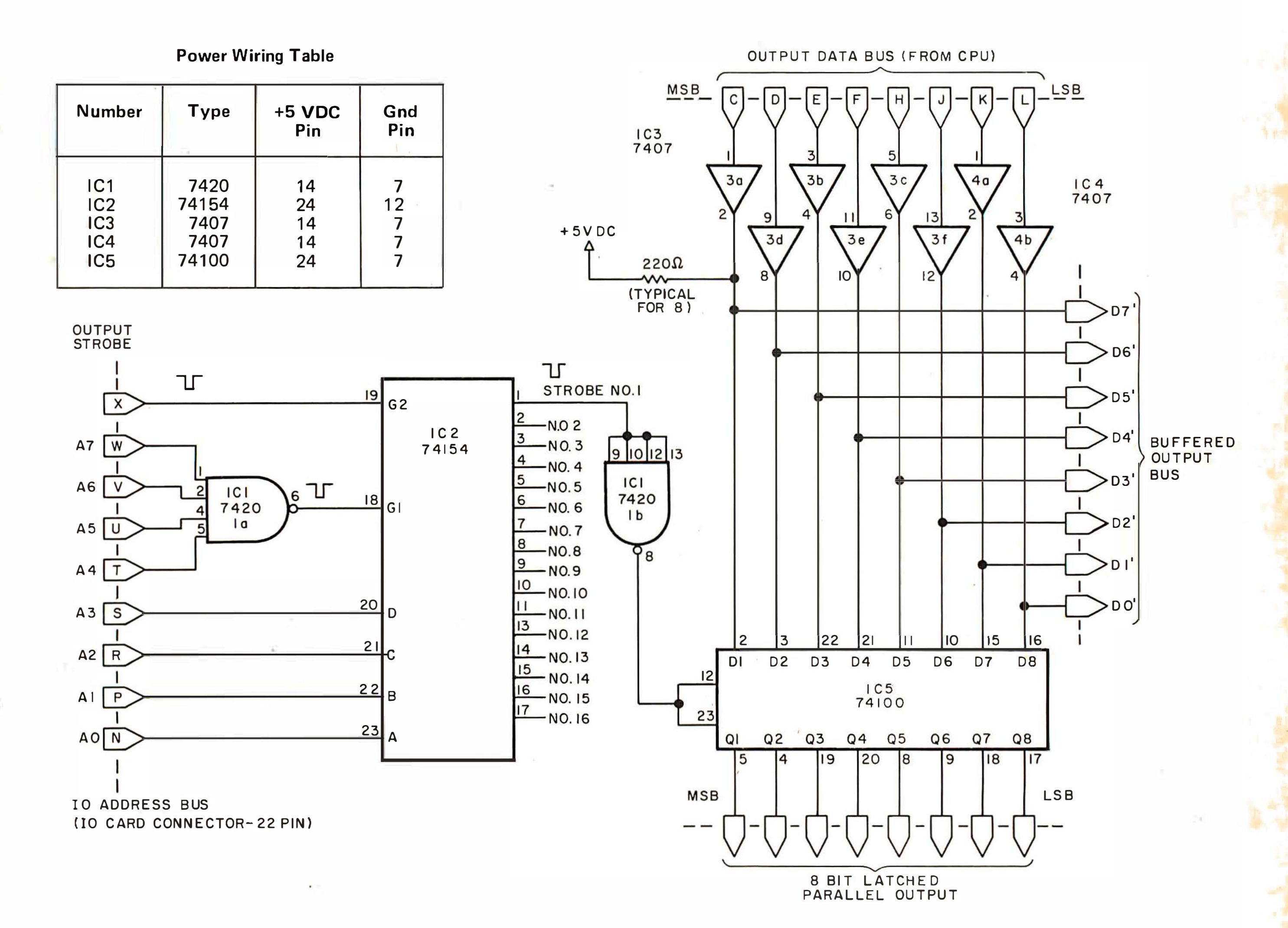
Steve Ciarcia
Box 582
Glastonbury CT 06033

I don't want to get into a fight over which microprocessor chip is better. They all have their favorable and unfavorable features. But, if you look a little closer, you may find that some of these extra features can be added with very little expense.

I was speaking with a fellow computer nut recently, and he was arguing about the merits of the 6800 versus the 8080. I really didn't care to continue the conversation nor to justify why I had an 8080 and Z-80. But, when he said that one reason was that the 6800 had memory mapped IO and the 8080 didn't, I knew he didn't know what it was.

This of course made me curious, and I approached a number of 8080 users to ask if they knew what memory mapped IO was. They assured me that they did, and that it was in fact one of the main features of the 6800. But such a feature is hardly exclusive to the 6800!

First of all, memory mapped IO means simply that a portion of memory address space has been reserved for interfacing with external devices. A byte of data is stored into a memory location, as always, but this storage unit, rather than being made up of 1024 bit programmable memory chips, is an



Power Wiring Table

Number	Type	+5 VDC Pin	Gnd Pin
IC1	7420	14	7
IC2	74154	24	12
IC3	7407	14	7
IC4	7407	14	7
IC5	74100	24	7

Figure 1: A schematic diagram for a direct addressed output port decoding circuit. The port assignments as diagrammed are from octal codes 360 to 377. The bus pin assignments are for the Digital Group bus system, but the Altair (S-100) bus is logically equivalent.

Intel delivers SDK-85. It's the quickest way to sink your teeth into 8085 design.

Intel wants you to prove to yourself why the 8085 has become the new industry standard microcomputer. To make it easy for you to do that, our System Design Kit for the 8085 is available now for only $250.

SDK-85 is the best way we know for you to evaluate MCS-85™ and develop prototypes of 8085-based designs, because it gives you a hands-on look at this important new microcomputer's capabilities.

And to simplify your evaluation, we've designed SDK-85 as a stand-alone kit. It comes complete with an integral keyboard for system control and data/program entry, and LED display output. To simplify programming, debugging and operation we've incorporated an on-board, ROM-resident software monitor.

The 8085 family of components provides you with unprecedented design flexibility. The basic three-chip, high level integration MCS-85 system is included in SDK-85. It includes the 8085 CPU, 8155 256-byte RAM with I/O and timer and 8355 2K-byte ROM with I/O. And there's an on-board single-chip keyboard/display interface, the 8279. Sockets are provided for easy RAM and ROM/EPROM expansion. And there's ample free space layed out for easy wire wrap expansion using Intel's complete family of programmable peripheral controllers and your own prototype logic and special circuitry.

SDK-85 makes an excellent teaching aid for both microprocessor design and programming courses, for microcomputer design seminars and as a project for the progressive hobbyist. Because the 8085 is the most advanced microcomputer, SDK-85 is the key to state-of-the-art knowledge.

SDK-85 can be assembled in just a few hours with a soldering iron and a few basic tools. Hook it up to your 5V power supply and it's operational the same day you receive it. You can get your SDK-85 from any of Intel's distributors for $250 in single unit quantities.

To order SDK-85 or any of the MCS-85 components, contact: Almac/Stroum, Components Specialties, Cramer, Hamilton/Avenet, Harvey Electronics, Industrial Components, Pioneer, Sheridan, L.A. Varah, Wyle Liberty/Elmar or Zentronics.

Or, for more information, use the reader service card or write: Intel Corporation, 3065 Bowers Avenue, Santa Clara, CA 95051. Telephone: (408) 246-7501.

intel delivers.



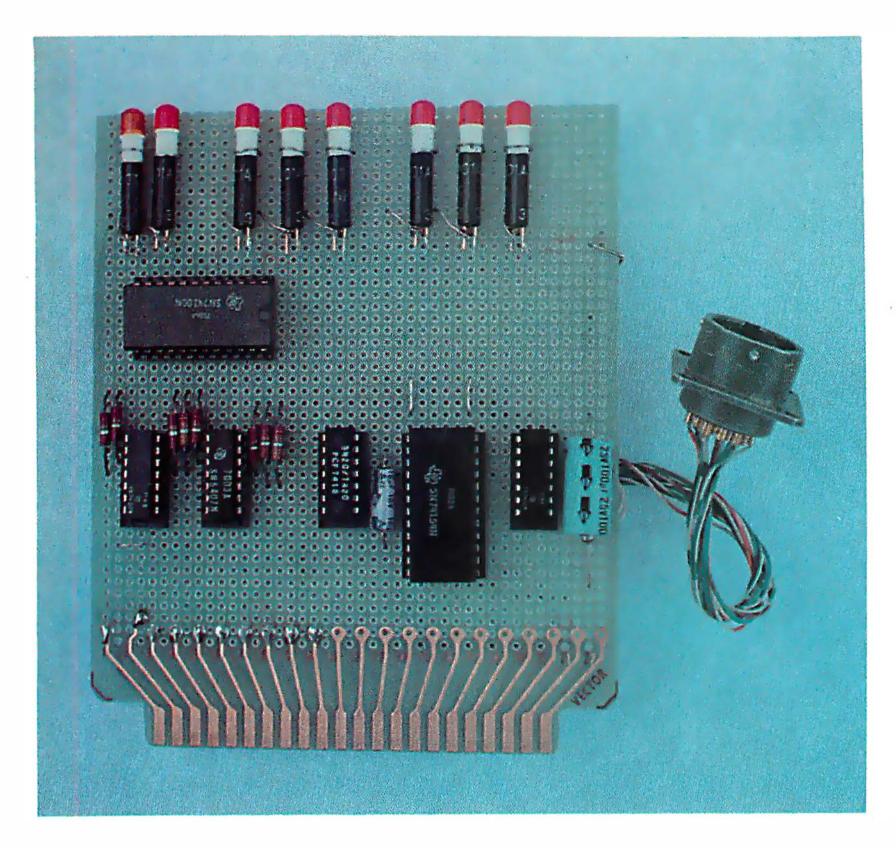


Photo 1: A realization of the hardware circuit shown in figure 1 with the addition of eight lights connected to the outputs of IC5. The connector attaches to bus lines for the author's other front end projects.

8 bit storage register such as a 74100. This type of procedure provides access to the data byte through the "back door," or output lines of the 74100. If you have followed me to this point, you can see that the concept of memory mapped IO is applicable to any microprocessor that directly addresses memory! I don't know of too many processors which operate without this ability, so we'll just have to conclude that any microprocessor can be wired to provide memory mapped IO, including the 8080.

Look no further! It's a bird. . .it's a plane. . .no, it's Superchip! It looks like an 8080, acts like an 8080 and, while not trying to steal Motorola's thunder, has memory mapped IO! The name of this new chip? Well, it's the plain old 8080 with an *intelligent user.*

Why should I consider memory mapped IO?

The 8080 directly addresses 64 K bytes of memory and 512 IO ports (256 in and 256 out). The only way data can arrive at an output port is by being passed through the accumulator and routed to a particular port by a 2 byte output instruction. Similarly, a 2 byte instruction directs input data to the accumulator. Additional programming is necessary to store this input byte in memory.

Photo 2: A prototype for the circuit shown in figure 2.

Obviously, if the data path went to a memory location instead of an output port, a broader range of instructions would be available. The 8080 (like most computers) has some very powerful instructions when it comes to memory operations. For the 8080 these include MOV, MVI, STAX and STA instructions which, by definition, are added to the output data manipulation repertoire with memory mapped IO.

Often the best way to approach a new subject is to analyze the present method. Figure 1 illustrates the basic design of an 8080 output "port." To emphasize simplicity I've used 74100 latches for this example rather than the more complex ports such as the Motorola 6820 peripheral interface adapter. This configuration provides 16 output strobes, starting with the octal output port address 360 and ending with octal 377. Integrated circuits 1 and 2 decode the address bus and, when provided with an output strobe during an output instruction, load the present contents of the data bus into an 8 bit storage register (IC5). ICs 3 and 4 provide buffering and allow more 74100s to be attached to the buffered ouput bus lines for multiple ports. The pin designations are for the Digital Group bus system, but the Altair (S-100) bus is logically equivalent.

Converting an output system to memory mapped IO (illustrated in figure 2) requires the addition of two more integrated circuits, ICs 6 and 7, to decode the additional eight lines associated with memory addressing. With the decoding arrangement illustrated in figure 2, the 16 output (memory) loca-

Power Wiring Table

Number	Type	+5 VDC Pin	Gnd Pin
IC1	7420	14	7
IC2	74154	24	12
IC3	7407	14	7
IC4	7407	14	7
IC5	74100	24	7
IC6	7402	14	7
IC7	7420	14	7

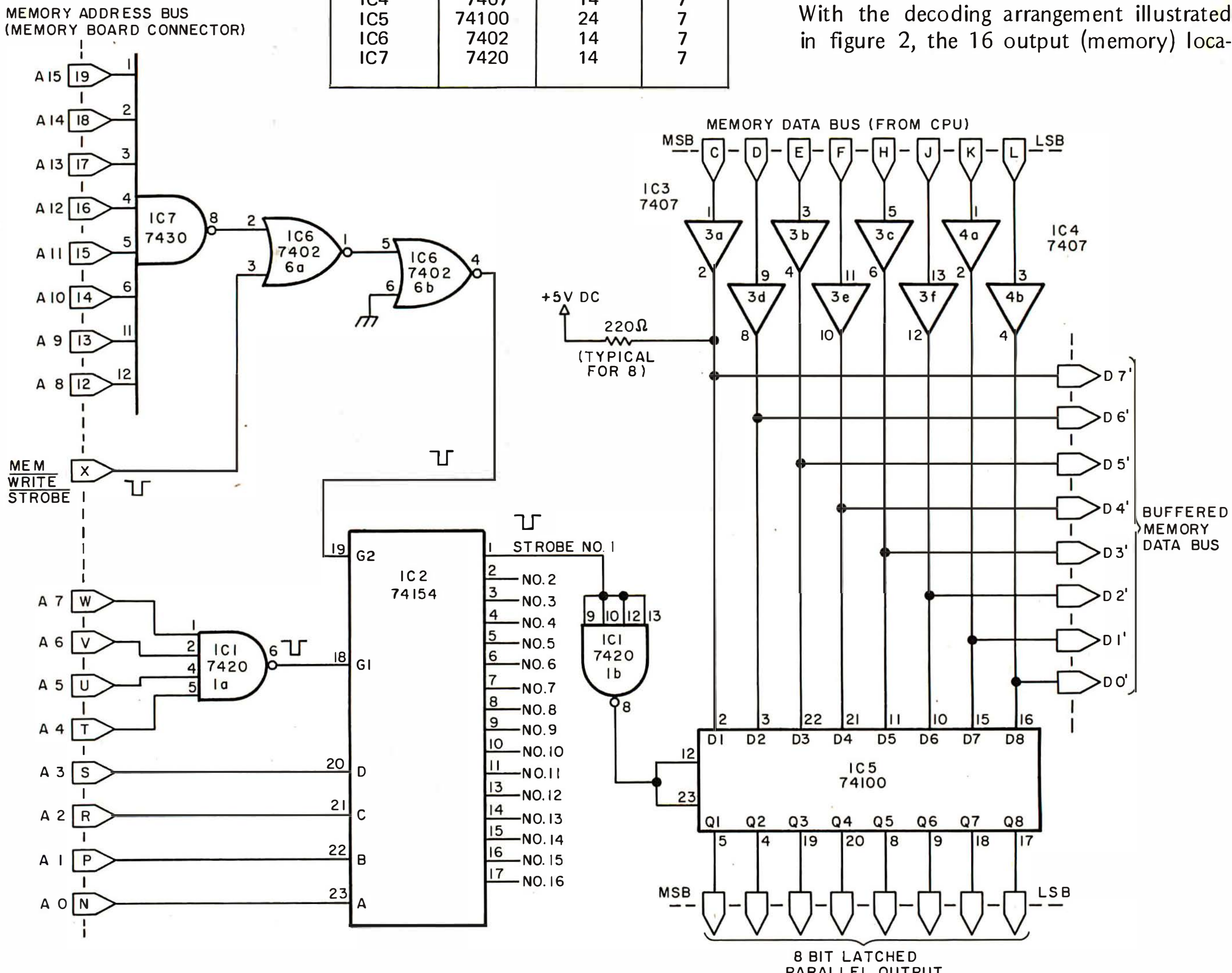


Figure 2: A schematic diagram for a memory addressed output port decoding circuit. The port assignments in this case are from split field octal memory addresses 377/360 to 377/377. Here again, the bus pin assignments are for the Digital Group bus.


RESET
EQUINOX 100

Example 1: Output the contents of the B register to port r.

- **8080 Direct IO**

```
MOV B, A     ; Move the contents of the B register to the accumulator
OUT r        ; Output the accumulator to port #r
             ; Total bytes    3
             ; Total states  15
```

- **Memory Mapped IO**

```
LXI HL       ; Set memory pointer HL
MOV M, B     ; Move B register to memory location HL
             ; Total bytes    4
             ; Total states  17
```

For simple data manipulations like this, the direct IO technique, which is familiar to all 8080 users, occupies less memory space.

Example 2: With two 8 bit digital to analog convertors attached to output registers, generate two sawtooth waveforms 180° out of phase.

- **8080 Direct IO**

```
          LXI BC       ; Load initial values into B and C (000, 200 octal)
CONTINUE  INC B        ; Increment the B value
          MOV B, A     ; Move the contents of the B register to the accumulator
          OUT 1        ; Output the accumulator to port 1 (1st sawtooth)
          INC C        ; Increment the C value
          MOV C, A     ; Move the contents of the C register to the accumulator
          OUT 2        ; Output the accumulator to port 2 (2nd sawtooth)
          JMP CONTINUE
                         Total bytes    14
                         Total states   60 (one pass)
```

- **Memory Mapped IO**

```
          LXI HL       ; Load initial values into H and L (000, 200 octal)
CONTINUE  INC H        ; Increment the H value
          INC L        ; Increment the L value
          SHLD ADDR; Store H and L in two consecutive memory locations wired
                         as output registers.
          JMP CONTINUE
                         Total bytes    11
                         Total states   46 (one pass)
```

tions will be from split octal addresses 377/360 to 377/377.

Now let's compare a couple of simple programs written using each method (see examples 1 and 2). It can be easily seen that the extra instuctions which operate on memory can greatly improve the output speed of the 8080. This extra speed, though not necessary when driving a 110 bps Teletype, can be a saving grace in a computer music or graphics application. In fact, many video display drivers utilize this technique.

Summary

There are certain advantages to converting 8080 peripherals to mapped versus direct IO. Among the major points to consider are the following:

- More IO ports are available. The full 64 K bytes of addressable memory space can be set up for IO. It is not inconceivable that a video graphics display will use 8 K bytes of memory. This, of course, means that the 8 K bytes are decoded to provide 8192 IO port assignments.
- Once the H and L registers have been loaded and provide a memory pointer, memory output is by 1 byte instructions (such as MOV and STAX).
- By not always having to pass through the accumulator, outputs are faster.
- 16 bit IO capability through the use of the LHLD and SHLD instructions.

Now, should you consider changing your 8080 system to memory IO? Frankly, if you are the type of person who will never write an assembly language program and is content to stick with high level languages such as extended BASIC, don't even consider it. If the software packages supplied by the computer manufacturers have worked consistently for you to this point, don't tempt fate. The majority of the systems sold, including Altair, IMSAI, DGS and so on, use 8080 IO instructions to all their peripherals. But many video systems bought as plug-in boards for the Altair (S-100) bus have memory mapped IO designs.

Delving into memory mapped IO should be reserved for people willing to use assembly language and prepared to modify standard software if required. In future editions I intend to investigate computer music applications where fast memory mapped 8080 (Z-80) IO will become a necessity. But, for the meantime, you should at least know what it is.■

Author's Note

I hope you've enjoyed the first installment of Ciarcia's Circuit Cellar. I'd like to have your comments and criticisms as well as any ideas you may have for future editions of this feature. I'm always interested in hearing from readers who have such brainstorms. Send all correspondence to Steve Ciarcia, POB 582, Glastonbury CT 06033, and please enclose a stamped, self-addressed envelope.

Simulation of Motion:

Stephen P Smith
POB 841
Parksley VA 23421

Part 1: An Improved Lunar Lander Algorithm

About the Author

Stephen P Smith's pet project as an amateur is a PASCAL compiler for a personal computer. Professionally, he leads the Computer Sciences Corporation support team attached to the range safety office at NASA Wallops Flight Center, where he and his team of analysts develop analytical methods and construct digital simulations of flight paths, flow fields and structural responses of rockets and aircraft. The BASIC programs which are part of this article and the remaining parts to come in several installments were developed and run on a Tektronix 4051, which uses a 6800 microprocessor and includes a BASIC interpreter.

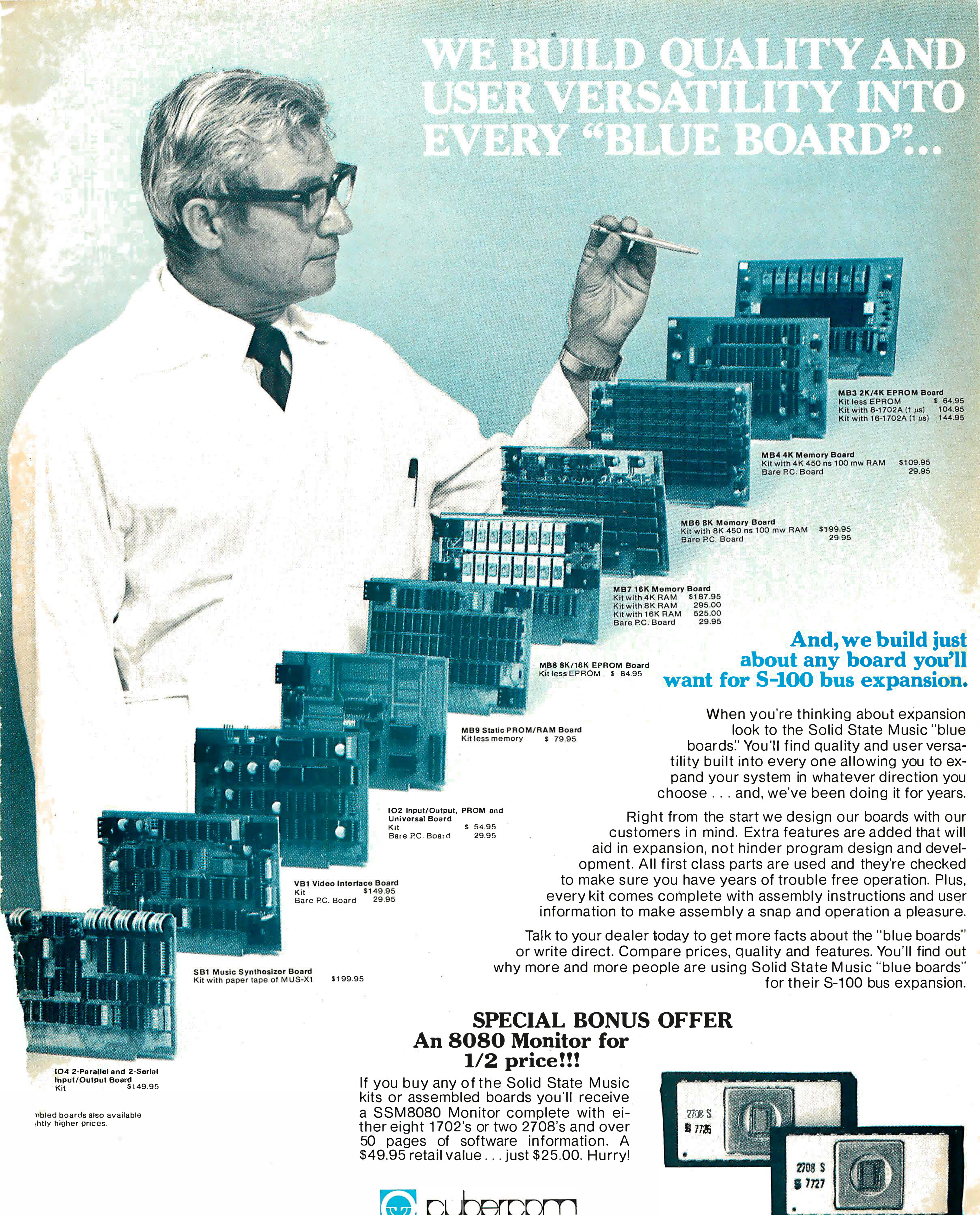
One of the most delightful applications for personal computers is games, not just playing them, but creating them. If you are like most enthusiasts, you will have begun with random number games like blackjack, but sooner or later you will want to work with games involving moving objects. To describe that motion using a microcomputer you will need to use a form of simulation. The simulation could involve detailed mathematical models solved with elegant numerical techniques.

More likely, the novice will begin by following the pattern of the simple lunar lander games which have appeared often in BYTE (see "Kim Goes to the Moon," by Butterfield in April 1977 BYTE, or "Controlling Small DC Motors with Analog Signals" by Dwyer, Critchfield and Sweer in September 1977 BYTE). The truly advanced simulations are best left to professionals with mainframe computer power, but the home user can progress well beyond the simple lunar lander game. By picking up the basic physics and simple numerical methods presented in this article and the following ones, you will learn to simulate a wide variety of motion. Whether you use these simulations to create games, like the real time LEM simulator presented here, or to develop new applications for your personal computer system, you will acquire some valuable additions to your applications software toolbox.

For any application involving motion, your simulation will be required to predict the speed and position of an object at some time in the future. The predictions can be made using a microcomputer if you first limit the type of motions considered at any point in the program. In the lunar lander game, for example, the excursion module (LEM) is only allowed to move up and down. The simulation is said to have one degree of freedom. Other degrees are possible, but the separation into different degrees of freedom is an important first step.

Let's see how a one degree of freedom simulation is performed. Thanks to Sir Isaac Newton and his apple (that was a fruit, not a computer), we know that an object will continue to move in any degree of freedom without changing speed until a force acts on it. To predict how the LEM will move, we need only to examine the forces which might be present and determine how they effect the up and down motion.

Because the moon has no atmosphere to involve us in aerodynamics, only two forces need be considered, gravity and thrust. Gravity makes the LEM fall faster. Thrust



Solid State Music
2102A Walsh Avenue
Santa Clara, CA 95050
(408) 246-2707

Circle 133 on inquiry card.

We're the blue boards

Generic Unit	Metric		
Length	1 meter	=	3.2808 feet
Velocity	1 meter per second	=	3.2808 feet per second
		=	2.2369 miles per hour
Acceleration	1 meter per second per second	=	3.2808 feet per scond per second
Mass	1 kilogram	=	2.2046 pounds (mass)
		=	0.0685 slugs (mass)
Force	1 newton	=	0.2248 pounds (force)

Table 1: This article was written using the metric system of units. As the front runners in an exciting new technical hobby, we should be more ready than most to accept the coming metric conversion in this country, but if you haven't been converted yet, the above table will be useful.

makes it fall more slowly. The exact effect of each can be calculated with only a few operations.

Gravity is the simpler of the two. It has exactly the same effect on every object. During each second of a lunar landing near the moon's surface, the moon's gravity will make a LEM fall 1.62 meters per second faster. (Those of you who wish to land on more exotic heavenly bodies are referred to table 2.) In most simulations, speed and position are considered positive if they are directed upward, in this case away from the lunar surface. To simulate 1 second of fall through lunar gravity we must subtract 1.62 meters per second from the present speed. If the LEM is moving at -100 meters per second now (100 m/sec downward), 1 second later it will be moving at -101.62 meters per second.

In many games, the effect of thrust is also simulated by a constant change in speed. Often it is given in multiples of gravity called "g"s. One "g" of thrust adds 1.62 meters per second to the speed, just as gravity subtracts that amount. Two "g"s add twice that, and so on. This assumption reduces the complexity of the simulation, but it fails to demonstrate the way in which forces actually cause changes in speed.

Unlike gravity, forces such as thrust do not have the same effect on every object. They have a larger effect on light objects than they have on heavier ones. It is important to consider this fact in accurate simulations, because weights can change. The LEM becomes lighter as it burns fuel to create thrust. A given value of thrust will have a larger effect toward the end of the flight than it will at the beginning.

Weight is not really the correct term to use when calculating that effect. We should talk instead of mass. The difference is subtle, but important. Mass is a basic property of matter. Weight is the result of gravity pulling on the mass. A man on the moon weighs only 1/5 as much as he does on earth, but his mass is the same. This is true because the moon's gravity pulls only 1/5 as strongly on his mass. The effect of a force is determined by the mass of an object, not by its weight. A given thrust will have the same effect on a LEM whether the LEM is landing on the moon, on earth, or is floating "weightless" in space.

In the metric system, the unit of mass is the kilogram. The unit of force is the newton. These units are very convenient for calculating the effect of a force on the motion of an object. The force (in newtons) divided by the mass (in kilograms) is exactly equal to the rate of change in speed ("acceleration" in meters per second per second). No additional constants are needed as they are when units of feet and pounds are used. For example, let our LEM have a mass of 1000 kg and let its engine produce a thrust of 10,000 newtons. To simulate 1 second of thrust, a program would add 10 meters per second to the speed (10000/1000) to account for 1 second's worth of acceleration.

Remember though that during the same second 1.62 meters per second must be subtracted to simulate the effect of gravitational acceleration. The actual change in speed will be 10.-1.62=8.38 meters per second. In two seconds, the change will be twice that or 16.76 meters per second. In half a second, the change will be one half as much and so on. While this may seem obvious, it illustrates an important point. The change that each force makes in the speed in 1 second may be determined separately. The separate effects are added up and then multiplied by the length of time we are simulating to find the actual value the simulation program will add to the speed.

Now that we can predict speed, let's apply the same technique to predict the

Heavenly Body	Surface Gravity (m/sec^2)	Heavenly Body	Surface Gravity (m/sec^2)
Moon	1.62	Asteroids	
Earth	9.80	Ceres	0.85
Mercury	3.95	Pallas	0.54
Venus	8.72	Juno	0.21
Mars	3.84	Vesta	0.43
Jupiter	23.16	Jupiter's moons	
Saturn	8.77	Ganymede	3.43
Uranus	9.46	Io	2.26
Neptune	13.66	Europa	1.98
Pluto	4.89	Callisto	3.20

Note that the gravitational accelerations shown in this table are surface accelerations, valid during the final stages of a landing when a spacecraft is relatively near the heavenly body. A more complicated simulation is required if movement far away from the heavenly body is contemplated.

Table 2: Players who grow adept at lunar landings may wish to try landing on some other heavenly bodies. The above table of accelerations due to gravity is provided for them.

We Have GOOD NEWS and BAD NEWS

The good news is that you can now add any S-100 bus compatible component to The VERSATILE CRT.

The bad news is you'll have to decide for yourself what components will combine to make the best system for you.

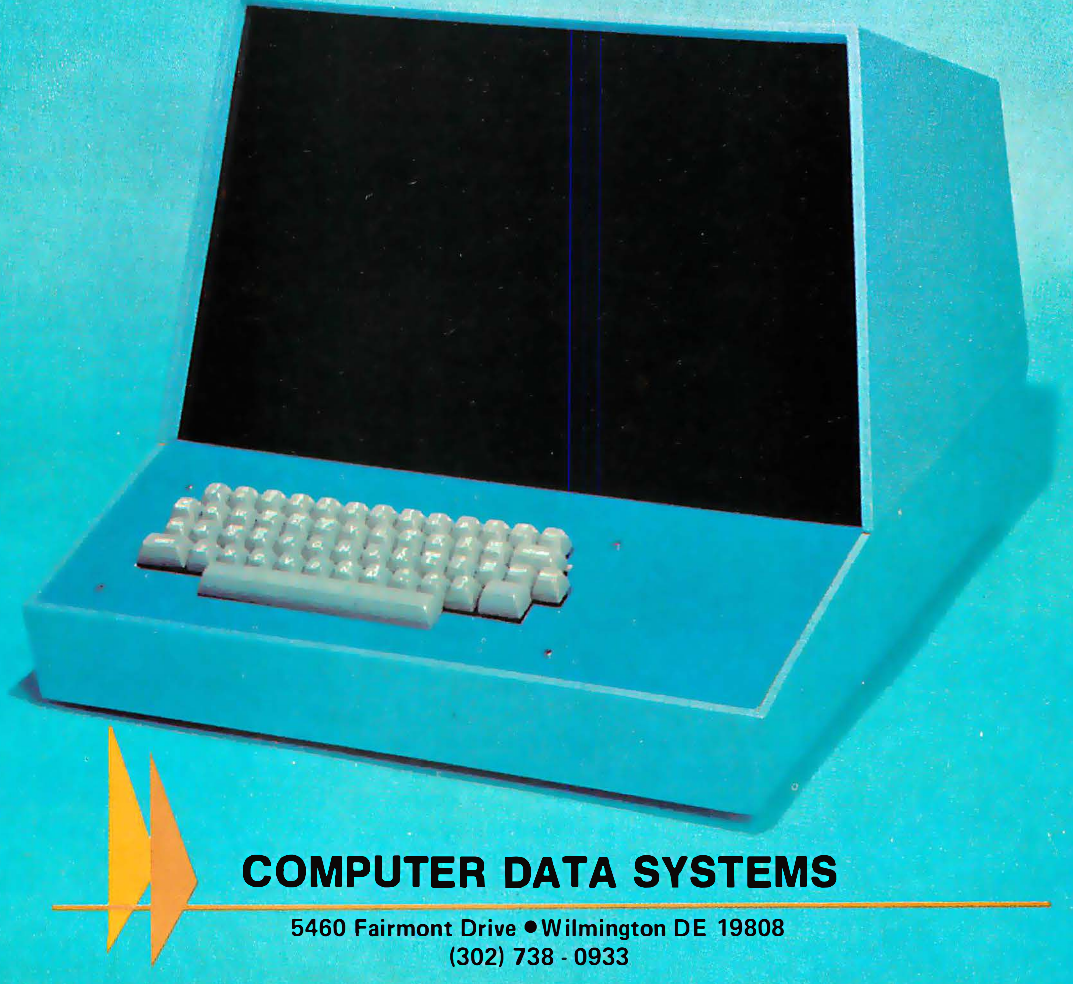
Just by looking you can see it's a rugged, professional unit with a 9'' video monitor covered with smoked plexiglass, and a 53-key ASCII keyboard.

But there's more than meets the eye. What you can't see is the mainframe which is a 10-slot plexiglass card rack. Or its fully shielded motherboard with 10 spaces provided for sockets (2 sockets are already included for you). Or the heavy duty filtered power supply and 75 CFM cooling fan. Or the power switch and cord on the rear panel. Or the space available for a floppy disk system. Or the expansion capabilities on the back panel for the addition of sockets.

You'll receive The VERSATILE CRT fully assembled with a 90-day warranty, about 20 days after we get your order. Or you can buy one at your nearest dealer. Dealers can have super fast delivery from stock to 20 days. You'll also receive complete documentation and operating manual. Total price for The VERSATILE CRT is $699.95.

Dealer inquiries are welcomed.

Compact and Expandable, The VERSATILE CRT lets you do it your own way.



COMPUTER DATA SYSTEMS

5460 Fairmont Drive • Wilmington DE 19808
(302) 738 - 0933

Circle 27 on inquiry card.

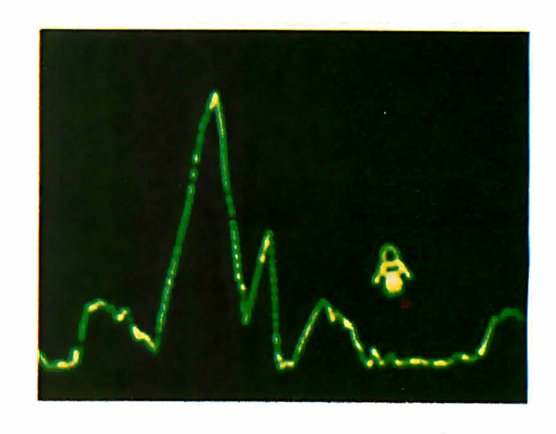


Photo 1: A scene from the "lunar lander" program which is the Digital Equipment Corportation's graphics equipment demonstration program. This simulation is a real time model of a lunar landing in which a light pen is used to input control information and displays track the landing. The object of the game is to land near (but not on) the only MacDonalds' hamburger stand on the moon. This simulation, like the one discussed in the article, has two degrees of freedom; superficially it differs from the program of this article largely in its incorporation of real time graphic display light pen control inputs and a model of the lunar terrain.

position. We have shown that if the LEM is moving downward at 100 meters per second now, (speed=-100) then in 2 seconds the speed will be -100.+2.x(THRUST/MASS -1.62). Similarly, if the LEM is 10000 meters above the moon now, in 2 seconds it will be 10000.+2.x(speed) meters up. Just as we multiply the forces by time and add the product to the speed, we multiply the speed by time, and add the product to the position.

What we have just done is to predict the speed and position at a "step" of 2 seconds into the future. In the jargon of simulation, 2 seconds is the step size. The step size can take any value you choose. Returning to the 1000 kg LEM, let the step size be 0.1 seconds. For a present speed of -100 meters per second, the speed predicted for 0.1 seconds in the future is -100.+0.1x(10000./1000.-1.62)=-99.16 meters per second. If the position now is 10000 meters, then the position predicted for 0.1 seconds in the future is 10000.+0.1x(-99.16)=9990.08 meters above the moon.

Using these values of speed and position we can find new values for the forces and mass. We can then step the simulation into the future once again. The process can continue indefinitely, but usually one or more variables is tested for an end condition at each step. The test might be on position (Are you still above the moon?), on mass (Is there fuel remaining?), or on some other variable. Should any of the tests fail, the program will branch and end the simulation.

Adding a New Degree of Freedom

You now know the basic procedure for simulating motion in one degree of freedom. The LEM simulation has been in one degree because we have only predicted the up and down movements. These are called vertical motions. Suppose that we also predict the way the LEM moves horizontally, in other words, from side to side. The pilot must not only reach the surface of the moon successfully, but also land close to his target. While the pilot's task has become more complicated, our simulation fortunately has not. Just as we are able to calculate the effects of each force separately, we are able to make calculations for speed and position separately in each degree of freedom.

To make those calculations for the second degree of freedom, first determine what forces are acting. Gravity, by definition, acts only up and down. It does not enter into the horizontal calculations. So far, thrust has also been limited to vertical action, but we can easily add a second thrust acting to the side. Positive horizontal thrust should cause the LEM to move left, while negative thrust moves it right.

Since there are no other forces to consider, the change in horizontal velocity (in meters per second) will be exactly equal to the horizontal thrust (in newtons) divided by the mass (in kilograms). This is, of course, the same equation used in the first or vertical degree of freedom. Similarly, the same equations used to calculate vertical speed and position will be used to calculate horizontal speed and position.

Return to the example used earlier, but also consider the horizontal motion. Let the LEM start 100 meters to the left of its target moving at 10 meters per second to the right. Generally motion to the left will be considered positive and to the right negative, so the horizontal speed is -10 meters per second. We found that during a step of 0.1 seconds the vertical speed changed from -100 to -99.16, and the position changed from 10000 to 9990.08. Quite apart from those calculations, we may set a horizontal thrust, say 5000 newtons, and find that during the same step the horizontal speed will become -10+0.1x (5000/1000) or -9.5 meters per second. The horizontal position will become 100.+0.1x(-9.5)=99.05 meters. After making these calculations, the simulation

Continued on page 216

they're here!

the new HEATHKIT® low-cost personal computing systems

The new VALUE-STANDARD in personal computing systems! Heathkit computers give you the power and performance to go wherever your imagination and programming prowess take you. They're designed to get you up and running fast, interface with I/O devices easily and quickly, accept additional memory and I/O devices, store and retrieve data with speed and accuracy, respond to your requests with lightning speed. They offer complete mass storage capabilities, power and reliability for any programming application, and they're priced low enough to give you real VALUE for your computer dollar! We've told you they're the ones you've been waiting for — here's why!

These Heathkit computer products are "total system" designs with powerful system software already included in the purchase price. They're the ones you need to get up and running fast. And they're backed by superior documentation including easy-to-follow step-by-step assembly and operations manuals, and service support from the Heath Company, the world's largest and most experienced manufacturer of electronic kits.

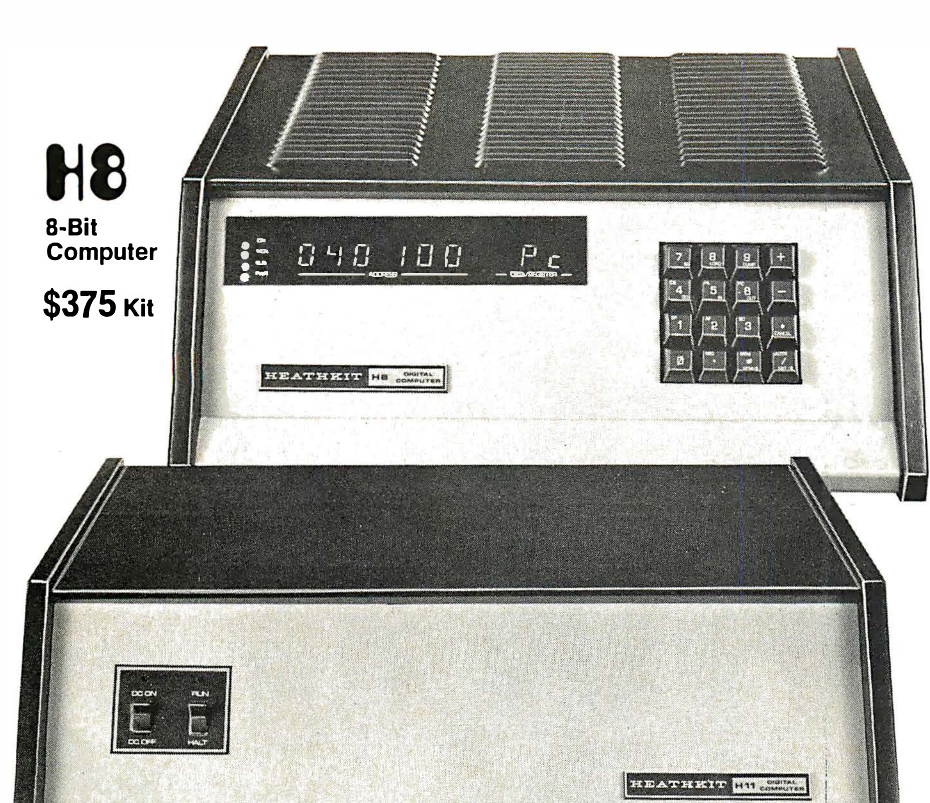
NEW H8 8-Bit Digital Computer. This 8-bit computer based on the famous 8080A microprocessor features a Heathkit exclusive "intelligent" front panel with octal data entry and control, 9-digit readout, a built-in bootstrap for one-button program loading, and a heavy-duty power supply with power enough for plenty of memory and interface expansion capability. It's easier and faster to use than other personal computers and it's priced low enough for any budget.

NEW H11 16-bit Digital Computer. The most sophisticated and versatile personal computer available today — brought to you by Heath Company and Digital Equipment Corporation, the world leader in minicomputer systems. Powerful features include DEC's 16-bit LSI-11 CPU, 4096 x 16 read/write MOS memory expandable to 20K (32K potential), priority interrupt, DMA operation and more. PDP-11 systems software for fast and efficient operation is included!

NEW H9 Video Terminal. A full ASCII terminal featuring a bright 12" CRT, long and short-form display, full 80-character lines, all standard serial interfacing, plus a fully wired and tested control board. Has autoscrolling, full-page or line-erase modes, a transmit page function and a plot mode for simple curves and graphs.

NEW H10 Paper Tape Reader/Punch. Complete mass storage peripheral uses low-cost paper tape. Features solid-state reader with stepper motor drive, totally independent punch and reader and a copy mode for fast, easy tape duplication. Reads up to 50 characters per second, punches up to 10 characters per second.

Other Heathkit computer products include a cassette recorder/player and tape for mass storage, **LA36 DEC Writer II keyboard printer terminal,** serial and parallel interfaces, software, memory expansion and I/O cards, and a complete library of the latest computer books. The Heath User's Group (HUG) provides a newsletter, software library and lots more to help you get the greatest potential from your Heathkit computer products. We've got everything you need to make Heath your personal computing headquarters, send for your FREE catalog today!



H8
8-Bit Computer
$375 Kit



Also Available: the famous LA36 DEC Writer II Keyboard Printer Terminal only $1495 (Assembled)!

H11
16-Bit Computer
$1295 Kit



H9
Video Terminal
$530 Kit



H10
Paper Tape Reader/Punch
$350 Kit

FREE!

HEATHKIT CATALOG

Read all about our exciting computer systems and nearly 400 other fun-to-build, money-saving electronic products in kit form.

Prices are mail-order FOB, Benton Harbor, Michigan.
Prices and specifications subject to change without notice.
Heath Company, Dept. 334-350 Benton Harbor, Mi 49022

HEATH
Schlumberger

Heath Company, Dept. 334-350
Benton Harbor, Michigan 49022

Please send me my FREE Heathkit Catalog.
I am not on your mailing list.

Name____

Address______________________________________

City________________________ State__________

CP-126 Zip__________

Circle 63 on inquiry card.



Photo 1: Students from the Kenosha area enjoy a computer display at the 1977 University of Wisconsin-Parkside Computer Fair.

A Minicomputer Fair: Tiny and Personal

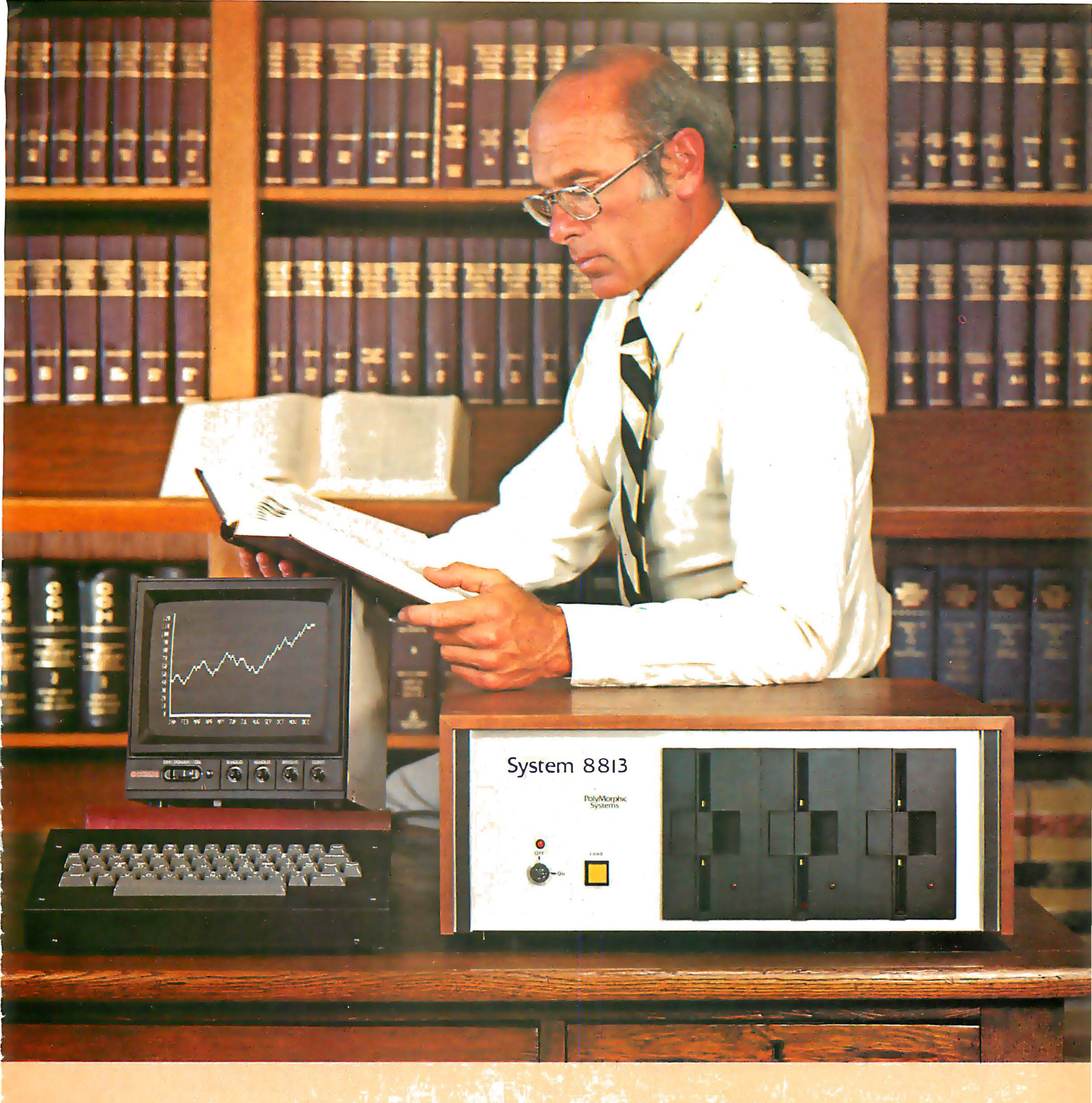
Donald T Piele
Assoc Professor of Mathematics
University of Wisconsin—Parkside
Kenosha WI 53140

If you start planning in April for a computer fair in June, you are probably either a novice, mini-minded, crazy, or all of the above. But sometimes a bit of insanity is just what is needed to make one jump in and do something new. Uncertain of what would happen, we plowed ahead with our fair, and we're glad we did.

The "us" I am referring to is the Center for the Application of Computers, a small group of faculty members at the University of Wisconsin-Parkside who share a common interest in computers and their many uses. We decided, rather late in the year, that an excellent way to proselytize our colleagues and generate interest among students and the general public would be to sponsor a computer fair. Our broad objective was to provide a forum for the rapidly developing field of personal computing with all its associated implications and applications.

Exhibits

A viable computer fair needs hardware exhibits. Unfortunately, Kenosha WI is not located in Silicon Gulch, and manufacturers cannot afford the time and money to attend every computer fair that springs up around the country. However, local computer stores, or those within a day's drive, are very interested in the exposure that such a fair brings. Despite the fact that Saturday is the busiest day of the week for them, we were able to line up six different computer stores for the fair, one as far away as Madison



The Computer for the Professional

Whether you are a manager, scientist, educator, lawyer, accountant or medical professional, the System 8813 will make you more productive in your profession. It can keep track of your receivables, project future sales, evaluate investment opportunities, or collect data in the laboratory.

Use the System 8813 to develop reports, analyze and store lists and schedules, or to teach others about computers. It is easily used by novices and experts alike.

Reliable hardware and sophisticated software make this system a useful tool. Several software packages are included with the machine: an advanced disk operating system supporting a powerful BASIC language interpreter, easy to use text editor, assembler and other system utilities. Prices for complete systems start at $3250.

See it at your local computer store or contact us at 460 Ward Dr., Santa Barbara, CA 93111, (805) 967-0468.

PolyMorphic Systems



Photo 2: Third graders Esther Marianyi (left) and Lisa Hanson from Southport and Roosevelt schools in Kenosha WI deep in thought as they program one of the computers at the Wisconsin Computer Fair.

(120 miles). Many manufacturers who were contacted but could not come helped out by encouraging the stores in our area to attend and display their products.

Other sources of hardware were hobbyists and computer clubs within driving range. Two Chicago area clubs and the Wisconsin Computer Society (an amateur computer club) were invited, and they responded with a number of excellent displays. Two $25 cash prizes were donated by BYTE magazine for the best "homecooking."

The support we received from the computer stores, clubs and a few local manufacturers made the hardware component of our fair very successful.

Manufacturers who could not come usually sent the all-important free brochures that everyone enjoys collecting at a fair whether they ever read them or not. A few generous manufacturers such as Vector, OK Tool and Hexadaisy included samples of their products which we could use as valuable door prizes.

Speakers

Another important component of every fair is the speakers. Throughout the day, a number of "small talks" (one half hour in length) were given by members of the Center for the Application of Computers, faculty members from other schools, hobbyists and students. Topics ranged from an introduction to personal computing, cryptography, microcomputers in the laboratory, and computer graphics, to optical character recognition and speech conversion. The featured speaker for the day was Ted Nelson, the writer, showman and computer guru who came armed with his talk, "The End of the Dinosaurs."

Programming Contest

The final component of our fair (and the one that made it very special) was the First Annual Interactive Computer Problem Solving Contest. The glitter of computer hardware with all its razzlers and dazzlers soon fades without an understanding of how one controls them through programming. Despite the fact that kids will sit for hours at a terminal playing a canned computer game, nothing can compare with the excitement that radiates from their faces when they successfully write their own programs to solve a problem.

The programming contest was divided into four categories: 1st thru 6th grade, 7th thru 10th grade, 11th thru 12th grade, and college. The contestants entered as teams of up to three members each and were assigned one terminal per team. Five problems of varying difficulty were handed out with a 2 hour time limit for solution. The 11th thru 12th category proved to be the most popular, and one 2 hour session with 19 teams was devoted exclusively to this category. After two hours each team turned in their solutions which consisted of a listing of the program and a sample run. The programs were quickly graded using the criterion of accuracy first and cleverness second.

The winners in the 11th thru 12th class were three seniors from Eau Claire WI (Tim Sirianni, Ellery Chan and Jeff Teeters) who traveled 300 miles that day to enter the contest. They did an outstanding job writing successful programs for all five problems within the 2 hour time limit—an exceptional performance surpassing even the college division that took the same exam. Prizes for first, second and third were awarded in all divisions, including trophies, books and complimentary subscriptions to publications.

Finally, the kids in the 1st thru 6th grade category deserve special attention. Earlier in the year, the special education class of K thru 4th graders from Kenosha Unified

schools, taught by Iris Helman and Sally Greenwood, had visited the computer center and played games on the terminals. This of course only whetted their appetites for more computer time, and arrangements were later made to return for four 1 hour lessons on programming in the BASIC language. Besides the mechanics of coding a computer, the elementary ideas of programming logic were emphasized through flowcharting. These ideas were discussed every day without a computer. The class enjoyed transforming its own scenario into a flowchart format using simple statements and branching conditions. We were pleasantly surprised at how entertaining and creative a flowchart can be when written by young children. The results were posted on a bulletin board at the fair and they proved to be a very popular attraction.

Future Plans

By starting earlier next year we hope to make the 2nd Annual UW-Parkside Computer Fair even more exciting. But quality, not quantity, will remain our long suit. About 700 attended the first fair and 1000 is our upper limit for a comfortable fair. Hardware exhibits will again be sought from local stores and vendors, but they will not play the dominant role that they do at larger fairs; talks and workshops exploring the expanding list of minicomputer applications will be just as important.

The 2nd Annual Interactive Computer Problem Solving Contest will be expanded and announced much earlier so that junior high and high schools throughout Wisconsin and Northern Illinois will have time to get ready. This year's exams will be freely handed out to schools along with instructions on how to run a computer problem solving contest locally. Through these contests we hope to lend encouragement to the growing number of teachers and young students who are eager to learn more about problem solving with the computer. In the process, we will be learning a great deal about this subject ourselves.

Finally, colleges and universities should take the lead in introducing the community they serve to the coming revolution of cheap computer power. They already have the physical resources to do the job with a minimum of cost. The return in public relations alone is worth much more than the investment. The local newspapers and Racine and Milwaukee television stations carried stories about the fair. Our fair represents one way of bringing computer awareness to the general public, and we highly recommend it.■



SPACE WAR DEFINITIONS

I have seen repeated mention, both in BYTE and in other sources, of the original computer game of Space War developed at MIT. What I have failed to see is any type of description or explanation pertaining to this classic king of computer games. What exactly does the original Space War entail in the way of display and participation? I am deeply interested in computer games, and I wonder just what was offered by this "oldie-goldie" to have rated such continued interest.

Again, in reference to MIT's Space War, are there currently any manufacturers' software or hardware products which are comparable? With thrillers like MiniTerm's Deluxe Space War and ECD's Animated Spacewar, I wonder if the current state of the art in computer games doesn't exceed that of the original MIT game.

Rick Craig
2609 E Woodlyn Way
Greensboro NC 27407

See the article by Dave Kruglinski on page 86 of the October 1977 BYTE for the answer to your question about what a classic Space War game does, illustrated by a practical example, which will probably not be the last such example seen in BYTE.

SAMPLING BIAS?

After reading your editorial in May 1977 BYTE I still find it hard to believe that only 1% of your readers are female. Did you by any chance look at marital status in the questionnaire? I would guess that in many cases both husband and wife are computer hobbyists. In most cases I would guess that married women interested in computers would share that interest with their husbands. The reverse however would not be as common. If my husband and I received your questionnaire he would most likely fill it out, thus skewing the results toward the 99% male figure. I'll bet what your survey really shows is the very small number of single women interested in computers and married women who are more interested than their husbands.

Next survey how about asking how many other people — other than the subscriber — read the magazine, and their age, relationship, level of interest, education, etc.

Looking at that 1% figure makes me feel very lonely. I'm sure there must be more women like myself who are interested in computers. I would enjoy hearing from other women hobbyists. Write and let me know who you are and what your interests are. I'll pass the information back to BYTE. It won't be an official survey, but I'll bet I'll get swamped with letters and postcards. Come on girls, let's show them that we exist!

Leah R O'Connor
6315 W Raven St
Chicago IL 60646

AN INVITATION TO ALL 1802 USERS: THE 1802 EXCHANGE

Very little software for the RCA CDP1802 is currently in the public domain. To remedy this situation I am going to publish a 10 page booklet listing available software. If you desire to sell or even give away your software please send me a listing for my review. My booklet will provide a complete description and cost information with a reference number corresponding to a number on an ordering coupon.

I plan to charge $1 for the booklet. This amount will also cover the costs associated with processing the coupons. The use of the coupon will reduce the costs to the person ordering from more than one source.

The publication date is set for early December 1977. Advance orders may be made at $1 per copy. Send all orders, software listings, and other correspondence to:

Ross Wirth
1636 S 108 East Av
Tulsa OK 74128

SELF-PROPAGATION MONSTERS

I discovered a real "bug" in the Z-80. If the registers are set up correctly, and the user has 64 K of programmable memory, a "living" creature can be created, similar to the interrupt driven monsters that pop up unexpectedly. Its only purpose in "life" is to procreate and eat food, namely time. If you object to my use of the term "life," go back and reread the definition. The "crea-

Rated G

The Best Game in Town.



Welcome to ComputerLand. An incredible adventure into the world of personal computers. A one-of-a-kind shopping experience.

Each ComputerLand store presents everything you ever wanted to know about computers. And then some.

Take our Game Room, for starters. You'll find excitement for the whole family in our endless variety of challenging computer games. You can battle the Klingons in an out-of-this-world game of Star Trek. Create an electronic work of art with a computer controlled TV. Test your skill in a game of computerized hangman.

You can even plot your biorhythm.

But we're more than just fun and games. Each ComputerLand store offers a knowledgeable and personable staff of professionals to serve you.

Plus the greatest available selection of micro components. Whether it's a data processing system for your business or a computer controlled sprinkler system for your home, you'll find whatever you need at ComputerLand.



Read on.

Genuine Service.

We want to supply you with the one system that's right. Rather than a complete system that isn't. Or a limited system that is.

That's why, at ComputerLand, you deal with real professionals who are also real people. People who speak your language . . . in addition to BASIC, COBOL or FORTRAN.

People, in short, who can offer both the novice and the old hand the same expert guidance in selecting the optimum system he or she needs.

Yet, assisting in the purchase is only the beginning of ComputerLand's service. If the kit you bought requires a little more do-it-yourself than you yourself can do, we provide assembly assistance.

If that complex program proves to be just that, we provide programming assistance.

And if your system breaks down, our in-store service department will get you back up and running.

Right now!

Great Selection.

Your first stop at ComputerLand may well be your last stop.

ComputerLand offers the finest quality and largest selection of all the major brand names. Like Apple Computer, Cromemco, DEC, Diablo, Hazeltine, ICOM, IMSAI, Lear Siegler, National Semiconductor, North Star, Texas Instruments, Vector Graphics and more.

Plus a complete inventory of tools, books and accessories.

What's more, at ComputerLand, we deal in product. Not promises. Our inventory is on our own shelves. Rather than the manufacturer's. So you can take delivery on tomorrow's components today.

Which means, simply put, that at ComputerLand, you get exactly what you want.

Exactly when you want it.



Be Our Guest.

Begin with the grand tour of our exhibit areas. "Test-drive" any of our individual systems.

Then tell us your needs. We'll sit down and talk about the system that's right for you. It's as easy as that at ComputerLand.

The great computer store. RATED G.

Call or write for the address of the ComputerLand store nearest you. Franchise opportunities available.



1922 Republic Avenue, San Leandro, CA 94577 (415) 895-9363

Circle 30 on inquiry card.

ture" is an ED B0 instruction, the block transfer LDIR.

To put it simply, it turns into a memory eater, copying itself everywhere into memory, prepetuating its existence. Externally, it looks like an ED B0 running loose. Here it is:

```
ORG 0000H

01 00 00   LD    BC,0000H   ;DO 65K TRANSFER
11 0B 00   LD    DE,000BH   ;UP 2 LOCATIONS
21 09 00   LD    HL,0009H   ;TO PERPETUATE

ED B0      LDIR             ;HERE'S THE ANIMAL
                            ;THAT GOES RUNNING OFF
```

Try it, it's fun.

Also in this category, there's the 14747 instruction in DEC PDP-11s. It copies itself lower in memory (even though DEC manuals say the instruction shouldn't work), and then executes the moved instruction! This one doesn't perpetuate, but it's neat to kill memory when you don't want someone to screw with some secret software.

Fred Beckhusen
MS 23
Mostek Corp
1215 W Crosby Rd
Carrollton TX 75006

Then of course there is the famous MVC instruction of the IBM 360 and 370 series, key to the famous OS 360 "time bomb" technique wherein a propagating MVC in supervisor mode mysteriously clears a 360's memory, crashing the machine hours after the joker who scheduled it has signed off TSO. Since the MVC moves 256 byte chunks and, once started, it always completes, the last MVC of the program goes one step further by clearing the program itself! (Reputedly, later than the mid 1970s, releases of IBM's TSO closed the holes by which clever programmers could get into supervisor mode from a TSO terminal.). . .CH

MOTOROLA EVALUATION KIT ARTICLES NEEDED

As an owner and user of an MEK-6800D2 kit from Motorola, I would like to see some software especially for this system with its J-BUG monitor. A somewhat similar but older system, the KIM-1, has a devoted following and many articles concerning this system have appeared in past BYTEs. I believe the D2 system, with a little encouragement, could also become popular. I know you are a 6800 fan, CH, so how about encouraging someone to write about this Motorola kit?

David Beach
POB 360
Frankford Ontario
CANADA K0K 2C0

PS: The MEK6800D2 appears fairly well thought out. Mine went together without any problems (I used sockets for all the chips, however.) and ran perfectly on the first power up.

MORE ON COMMERCIAL RADIO AUTOMATION

Joe Alwin's request in the February 1977 BYTE for information on microprocessor based radio automation systems is easily answered. McCurdy Radio of 108 Carnforth Rd, Toronto CANADA, has an 8080 based system that will do just what he wants. Data input is via keyboard or standard audio cartridges or cassettes for compatibility with other radio station equipment. Logging may be on Teletype, or the data may be recirculated in memory and used again for another day's programs. Data is displayed on a CRT.

And now perhaps one of your readers can help me. I am looking for a "Universal Alarm Annunciator." If any one of, say, 100 terminals is grounded, I want to display a one line alarm message on a CRT, eg: "#54: XMTR OFF AIR." The messages must be previously entered from a keyboard and must of course be protected against power failure. An additional "HELP" routine could be used to call up (off disk) a whole page of previously entered text describing what to do to solve the #54 alarm problem. As you will appreciate, the difficulty lies in solving the sorting problem economically. Including the CRT, keyboard and microprocessor, the whole thing should come in at less than $15,000. Has anyone such an item up their sleeve?

M Barlow
5052 Chestnut Av
Pierrefonds
Montreal CANADA

LORAN-C CLARIFIED

In the July 1977 BYTE, there was a letter from Ian McNicol in which there occurred a sort of throw-away line: " . . . why use OMEGA when there are satellite systems like LORAN-C?" Well, perhaps this is a pertinent question, but it displays a little misinformation. LORAN-C is not a satellite system. LORAN-C is a system consisting of a master station and two to four slave stations which broadcast a series of pulses which modulate a 100 kHz carrier. The master sends a signal which is received by the slave stations and the navigation receiver. The slave stations delay the master signal and rebroadcast it to the navigator. The LORAN-C receiver measures the time difference between arrival of the master and slave

Continued on page 145

UP AND RUNNING

TDL EQUIPMENT USED BY NEW JERSEY PUBLIC TELEVISION TO PROCESS NEW JERSEY GUBERNATORIAL PRIMARY ELECTION RETURNS

John Montagna, computer engineer (above left), lead this successful network team in generating election results speedily, efficiently and reliably using predominantly TDL hardware and software. Montagna created three programs to get the job done. The text for a SWAPPER program was written and assembled using the TDL TEXT EDITOR and Z80 RELOCATING MACRO ASSEMBLER. The SWAPPER text and all debugging was run through TDL's ZAPPLE MONITOR. The relocatable object code was punched onto paper tape. A MAIN USERS program updated votes and controlled air display. An ALTERNATE USERS program got hard copy out and votes in. The latter two programs were written in BASIC. Montagna modified the ZAPPLE BASIC to permit time-sharing between the two USERS programs.

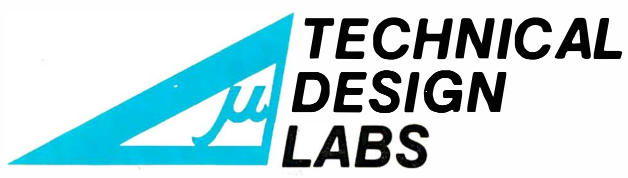
Four screens were incorporated, two terminals entered votes as they came in and were used to call back votes to check accuracy. Montagna called on the power and flexibility offered by TDL's ZPU board and three Z-16 Memory boards.

Montagna's setup worked constantly for over four hours updating and displaying state-wide and county-wide results without flaw.

"I chose TDL because they have all the software to support their hardware, and it's good; it has the flexibility to do the job."
John Montagna

We salute John Montagna and NEW JERSEY PUBLIC BROADCASTING for spearheading the micro-computer revolution.

TDL's XITAN SYSTEMS have the capacity to do similar tasks for you. Write to us for XITAN information and the name of your nearest TDL dealer.



RESEARCH PARK BLDG. H 1101 STATE ROAD
PRINCETON, NEW JERSEY 08540 (609) 921-0321

Circle 145 on inquiry card.


Introducing Apple II.™

You've just run out of excuses for not owning a personal computer.

Clear the kitchen table. Bring in the color TV. Plug in your new Apple II,* and connect any standard cassette recorder/player. Now you're ready for an evening of discovery in the new world of personal computers.

Only Apple II makes it that easy. It's a complete, ready to use computer, not a kit. At $1298, it includes video graphics in 15 colors. It includes 8K bytes ROM and 4K bytes RAM—easily expandable to 48K bytes using 16K RAMs (see box). But you don't even need to know a RAM from a ROM to use and enjoy Apple II. For example, it's the first personal computer with a fast version of BASIC permanently stored in ROM. That means you can begin writing your own programs the first evening, even if you've had no previous computer experience.

The familiar typewriter-style keyboard makes it easy to enter your instructions. And your programs can be stored on—and retrieved from—audio cassettes, using the built-in cassette interface, so you can swap with other Apple II users.

You can create dazzling color displays using the unique color graphics commands in Apple BASIC. Write simple programs to display beautiful kaleidoscopic designs. Or invent your own games. Games like PONG—using the game paddles supplied. You can even add the dimension of sound through Apple II's built-in speaker.

But Apple II is more than an advanced, infinitely flexible game machine. Use it to teach your children arithmetic, or spelling for instance. Apple II makes learning fun. Apple II can also manage household finances, chart the stock market or index recipes, record collections, even control your home environment.

Right now, we're finalizing a peripheral board that will slide into one of the eight available motherboard slots and enable you to compose music electronically. And there will be other peripherals announced soon to allow your Apple II to talk with another Apple II, or to interface to a printer or teletype.

Apple II is designed to grow with you as your skill and experience with computers grows. It is the state of the art in personal computing today, and compatible upgrades and peripherals will keep Apple II in the forefront for years to come.

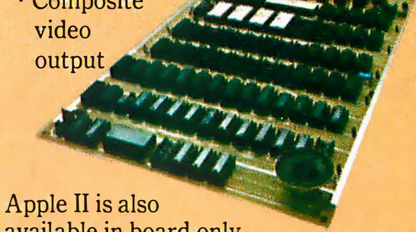
Write us today for our detailed brochure and order form. Or call us for the name and address of the Apple II dealer nearest you. (408) 996-1010. Apple Computer Inc., 20863 Stevens Creek Boulevard, Bldg. B3-C, Cupertino, California 95014.



Apple II™ is a completely self-contained computer system with BASIC in ROM, color graphics, ASCII keyboard, lightweight, efficient switching power supply and molded case. It is supplied with BASIC in ROM, up to 48K bytes of RAM, and with cassette tape, video and game I/O interfaces built-in. Also included are two game paddles and a demonstration cassette.

SPECIFICATIONS

- **Microprocessor:** 6502 (1 MHz).
- **Video Display:** Memory mapped, 5 modes—all Software-selectable:
  - Text—40 characters/line, 24 lines upper case.
  - Color graphics—40h x 48v, 15 colors
  - High-resolution graphics—280h x 192v; black, white, violet, green (12K RAM minimum required)
  - Both graphics modes can be selected to include 4 lines of text at the bottom of the display area.
  - Completely transparent memory access. All color generation done digitally.
- **Memory:** up to 48K bytes on-board RAM (4K supplied)
  - Uses either 4K or new 16K dynamic memory chips
  - Up to 12K ROM (8K supplied)
- **Software**
  - Fast extended integer BASIC in ROM with color graphics commands
  - Extensive monitor in ROM
- **I/O**
  - 1500 bps cassette interface
  - 8-slot motherboard
  - Apple game I/O connector
  - ASCII keyboard port
  - Speaker
  - Composite video output



Apple II is also available in board-only form for the do-it-yourself hobbyist. Has all of the features of the Apple II system, but does not include case, keyboard, power supply or game paddles. $598.

PONG is a trademark of Atari Inc.
*Apple II plugs into any standard TV using an inexpensive modulator (not supplied).

apple computer inc.™

Order your Apple II now.

from any one of the following authorized dealers:

ALABAMA
Computerland
3020 University Dr. N.W.
Huntsville 539-1200
The Computer Center
303 B. Poplar Place
Birmingham 942-8567

ALASKA
Team Electronics
Country Village Shopping Center
700 E. Benson Blvd.
Anchorage 276-2923
Team Electronics
404 E. Fireweed Lane
Anchorage 272-4823
Team Electronics
1698 Airport Way
Fairbanks 456-4157

ARIZONA
Byte Shop
813 N. Scottsdale Road
Tempe 894-1193

CALIFORNIA
Computer Components
5848 Sepulveda Blvd.
Van Nuys 786-7411
Computerland
11074 San Pablo Ave.
El Cerrito 233-5010
Computerland
22634 Foothill Blvd.
Hayward 538-8080
Computerland
6840 La Cienega Blvd.
Inglewood 776-8080
Computerland
24001 Via Fabricante
Mission Viejo 770-0131
Computerland
4233 Convoy Street
San Diego 560-9912
Computerland
117 Fremont St.
San Francisco 546-1592
Computerland
104 W. First Street
Tustin 544-0542
Byte Shop
6041 Greenback Lane
Citrus Heights 961-2983
Byte Shop
2233 El Camino Real
Palo Alto 327-8080
Byte Shop
496 S. Lake Ave.
Pasadena
Byte Shop
2626 Union Avenue
San Jose 377-4685
Byte Shop
1200 W. Hillsdale Blvd.
San Mateo 341-4200
Byte Shop
3400 El Camino Real
Santa Clara 249-4221
Byte Shop
2989 N. Main Street
Walnut Creek 933-6252
A-VIDO Electronics
2210 Bellflower Road
Long Beach 598-0444
Computer Country
2232 Salt Air Drive
Santa Ana 832-9681
Computer Playground
6789 Westminster Avenue
Westminster 898-8330
Computer Store
1093 Mission St.
San Francisco 431-0640
Electric Brain
3038 N. Cedar Ave.
Fresno 227-8479
Rainbow Computing, Inc.
10723 White Oak
Granada Hills 360-2171
Strawberry Electronics
71 Glenn Way #9
Belmont 595-0231

COLORADO
Byte Shop
3464 S. Acoma St.
Englewood 761-6232
Team Electronics
3275 28th Street
Boulder 447-2368
Team Electronics
The Citadel
Colorado Springs 596-5566
Team Electronics
107 S. College
Fort Collins 484-7500
Team Electronics
Teller Arms Shopping Center
2401 North Avenue
Grand Junction 245-4455
Team Electronics
2045 Greeley Mall
Greeley 356-3800

COLORADO (continued)
Team Electronics
1450 Main Street
Longmont 772-7800
Team Electronics
1022 Constitution Road
Belmont Plaza
Pueblo 545-0703

FLORIDA
Byte Shop
1044 E. Oakland Park Blvd.
Ft. Lauderdale 561-2983
Byte Shop
7825 Bird Road
Miami 264-2983

GEORGIA
Data Mart, Inc.
3001 N. Fulton Drive
Atlanta 233-0532

HAWAII
Real Share
190 S. King Street #890
Honolulu 536-1041

ILLINOIS
Team Electronics
Meadowdale Drive, Space 1A
Carpentersville 428-6474
Team Electronics
Northgate Mall Shopping Center
Decatur 877-2774
Team Electronics
Sandburg Mall
1150 W. Carl Sandburg Drive
Galesburg 344-1300
Team Electronics
Southpark Shopping Center
4200 16th Street
Moline 797-8261
Team Electronics
4700 Block — N. University Ave.
Peoria 692-2720
Team Electronics
1714 Fifth Avenue
Rock Island 788-9595
Team Electronics
321 N. Alpine Road
Rockford 399-2577
Team Electronics
Woodfield Mall F-119
Schaumburg 882-5864
Team Electronics
2716 S. MacArthur Blvd.
Springfield 525-8637
Computerland
50 E. Rand Road
Arlington Heights 255-6488
Computerland
9511 N. Milwaukee Ave.
Niles 967-1714
Itty Bitty Machine Company
1316 Chicago Avenue
Evanston 328-6800

INDIANA
The Data Domain
2805 E. State Blvd.
Fort Wayne 484-7611
The Data Domain
7027 Michigan Road
Indianapolis 251-3139

IOWA
Team Electronics
202 Main Street
Ames 232-7705
Team Electronics
Duck Creek Plaza
Bettendorf 355-7013
Team Electronics
4444 First Avenue N.E.
Lindale Plaza
Cedar Rapids 393-8956
Team Electronics
320 Kimberly Road
Northpark Shopping Center
Davenport 386-2588
Team Electronics
2300 Kennedy Road
Dubuque 583-9195
Team Electronics
Room 120 — Space 18
The Mall
Iowa City 338-3681
Team Electronics
2015 E. Fourth Street
Sioux City 252-4507
Team Electronics
K-D Stockyards Station
2001 Leech Avenue
Sioux City 277-2019
Team Electronics
2750 University Avenue
Waterloo 235-6507

KANSAS
Team Electronics
215 W. Kansas Avenue
Garden City 276-2911
Team Electronics
14 S. Main Street
Hutchinson 662-0632

KANSAS (continued)
Team Electronics
2319 Louisiana Street
Lawrence 841-3775
Team Electronics
1132 Westloop Shopping Center
Manhattan 539-4636
Team Electronics
Space 81-A Mid-State Mall
Salina 827-9361
Team Electronics
907 W. 27th Street Terrace
Topeka 267-2200
Team Electronics
Towne East Square
7700 E. Kellogg
Wichita 685-8826
Team Electronics
791 N. West Street
Wichita 942-1415
Team Electronics
"The Mall" on Harry Street
Wichita 682-7559
Barney & Associates
425 N. Broadway
Pittsburg 231-1970

KENTUCKY
Computerland
813 B Lyndon Lane
Louisville 425-8308
The Data Domain
506½ Euclid Avenue
Lexington 233-3346
The Data Domain
3028 Hunsinger Lane
Louisville 456-5242

MARYLAND
Computerland
16065 Frederick Road
Rockville 948-7676

MASSACHUSETTS
The Computer Store, Inc.
120 Cambridge Street
Burlington 272-8770

MICHIGAN
Team Electronics
Delta Plaza Shopping Center
Escanaba 786-3911
Team Electronics
M&M Plaza
Menominee 864-2213

MINNESOTA
Team Electronics
Ridgedale Mall
12503 Wayzata Blvd.
Minnetonka 544-7412
Team Electronics
204 Southdale Center
Edina 920-4817
Team Electronics
1248-50 Eden Prairie Center
Eden Prairie 941-8901
Team Electronics
207 Third Street
Bemidji 751-7880
Team Electronics
Kandi Mall South Hwy 71
Willmar 235-2120
Team Electronics
Crossroads Shopping Center
St. Cloud 253-8326
Team Electronics
Cedar Mall
Owatonna 451-7248
Team Electronics
Mesahi Mall
Hibbing 263-8200
Team Electronics
Thunderbird Mall
Virginia 741-5919
Team Electronics
Apache Plaza
Silver Lake Road
St. Anthony 789-4368
Team Electronics
1733 S. Robert Street
West St. Paul 451-1765
Team Electronics
2640 Hennepin Avenue S.
Minneapolis 377-9840

MINNESOTA (continued)
Team Electronics
455 Rice Street
St. Paul 227-7223
Team Electronics
110 Sixth Avenue S.
St. Cloud 251-1335
Team Electronics
6413 Lyndale Avenue S.
Minneapolis 869-3288
Team Electronics
1311 Fourth St. S.E.
Minneapolis 378-1185
Team Electronics
Maplewood Plaza
3000 White Bear Avenue
Maplewood 777-3737
Team Electronics
Madison East
Mankato 387-7937
Team Electronics
310 Grant Avenue
Eveleth 749-8140
Team Electronics
Har Mar Mall
2100 N. Snelling Avenue
St. Paul 636-5147
Computer Depot
3515 W. 70th Street
Minneapolis 927-5601

MISSOURI
Team Electronics
Biscayne Mall
301 Stadium Blvd.
Columbia 445-4496
Electronic Components Intl.
1306-B South Hwy 63
Columbia 443-5225

MONTANA
Team Electronics
613 Central Avenue
Great Falls 852-3281
Team Electronics
1208 W. Kent
Missoula 549-4119
Computers Made Easy
415 Morrow
Bozeman 586-3065

NEBRASKA
Team Electronics
148 Conestoga Mall
Highway 281 & 13th Street
Grand Island 381-0559
Team Electronics
2055 "O" Street
Lincoln 435-2959
Team Electronics
304 S. 72nd Street
Cedarnole Shopping Center
Omaha 397-1666
Team Electronics
Bel Air Plaza
12100 W. Center Road
Omaha 333-3100
Team Electronics
Sunset Plaza Shopping Center
Norfolk 379-1161
Team Electronics
The Mall
1000 S. Dewey
North Platte 534-4645

NEW HAMPSHIRE
Computermart
170 Main Street
Nashua 883-2386

NEW JERSEY
Computerland
2 De Hart Street
Morristown 539-4077
Computermart
501 Route 27
Iselin 283-0600

NEW YORK
Computerland
1612 Niagara Falls Blvd.
Buffalo 836-6511
Computerland
225 Elmira Road
Ithaca 277-4888
Co-op Electronics
9148 Main Street
Clarence 634-2193

NORTH CAROLINA
Byte Shop
1213 Hillsborough St.
Raleigh 833-0210

NORTH DAKOTA
Team Electronics
2304 E. Broadway
Bismarck 223-4546
Team Electronics
West Acres Shopping Center
Fargo 282-4562
Team Electronics
1503 11th Avenue N.
Grand Forks 746-4474
Team Electronics
209 11th Avenue S.W.
Minot 852-3281
Team Electronics
109 Main Street
Williston 572-7631

OHIO
The Data Domain
1932 Brown Street
Dayton 223-2348

OKLAHOMA
Team Electronics
1105 Elm Street
Stubbeman Village
Norman 329-3456
Team Electronics
Crossroads Mall
7000 Crossroads Space 2010
Oklahoma City 634-3357
Team Electronics
Penn Square Shopping Center
Penn Square
Oklahoma City 848-5573
Team Electronics
1134 Hall of Fame Avenue
Stillwater 377-2050
Team Electronics
5305 E. 41st
Southroads Mall
Tulsa 633-4575
Team Electronics
Woodland Hills Mall
7021 Memorial
Tulsa 252-5751
Team Electronics
Surrey Hills
Yukon 373-1994
Bits, Bytes & Micros
1186 N. MacArthur Blvd.
Oklahoma City 947-5646
High Technology
1020 W. Wilshire Blvd.
Oklahoma City 842-2021

OREGON
Team Electronics
1913 N.E. Third Street
Bend 389-8525
Team Electronics
1023 S.W. 1st
Canby 266-2539
Team Electronics
2230 Fairground Road N.E.
Salem 364-3278

SOUTH DAKOTA
Team Electronics
402 W. Sioux Avenue
Pierre 224-1881
Team Electronics
1101 Omaha Street
Rapid City 343-8363
Team Electronics
613 W. 41st Street
Sioux Falls 336-3730
Team Electronics
41st Street & Western Avenue
Western Mall
Sioux Falls 339-1421
Team Electronics
Sioux Empire Mall
4001 West 41st Street
Sioux Falls 339-2237
Team Electronics
223 Ninth Avenue S.E.
Watertown 886-4725

TEXAS
Byte Shop
3211 Fondren
Houston 977-0664
Computerland
6439 Westheimer
Houston 997-0909
Computer Shops, Inc.
13933 North Central
Dallas 234-3412
The Computer Shop
6812 San Pedro
San Antonio 828-0553
Computer Terminal
2101 Myrtle St.
El Paso 532-1777
The KA Computer Store
1200 Majesty Drive
Dallas

VIRGINIA
Home Computer Center
2927 Virginia Beach Blvd.
Virginia Beach 486-1700
Timberville Electronics
P.O. Box 202
Timberville 896-8926

WASHINGTON
Team Electronics
423 W. Yakima
Yakima 453-0313

WISCONSIN
Team Electronics
3209 Rudolph Road
Eau Claire 834-0328
Team Electronics
3365 E. Clairmont Parkway
Eau Claire 834-1288
Team Electronics
3365 E. Washington Avenue
Madison 244-1339
Team Electronics
7512 W. Appleton Avenue
Milwaukee 461-7800
Team Electronics
3701 Durand Avenue
Elmwood Plaza Shopping Center
Racine 554-8505
Team Electronics
3347 Kohler Avenue
Memorial Mall, Space H-4
Sheboygan 458-8791
Team Electronics
5300 S. 76th
Southridge Center
Greendale 421-4300
Team Electronics
Sunrise Plaza
Highway 8 East
Rhinelander 369-3900
Team Electronics
1505 Losey Blvd. S.
Village Shopping Center
LaCrosse 788-2250
Team Electronics
2207 Grand Avenue
Wausau 842-3364
Team Electronics
3301-3500 S. 27th Street
Southgate Shopping Center
Milwaukee 672-7600
Team Electronics
2619 Milton Avenue
Janesville 756-3150
Team Electronics
1801 Marshall Street
Manitowoc 684-3393
Team Electronics
7700 W. Browndeer Road
Northridge Center
Milwaukee 354-4880
Team Electronics
396 Park Avenue
Oshkosh 233-7050

WYOMING
Team Electronics
Hilltop Shopping Center
207 S. Montana
Casper 235-6691

CANADA
Future Byte
2274 Rockland
Montreal, Que. 731-4638

If you would like to be an Apple dealer, call Gene Carter, Director of Dealer Marketing, 408-996-1010.

apple computer inc.™

20863 Stevens Creek Boulevard, B3-C
Cupertino, California 95014
(408) 996-1010

Circle 4 on inquiry card.

BYTE's Bugs

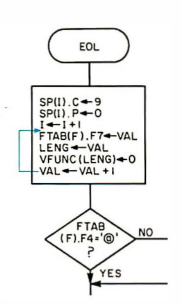
Some Comments on "An APL Interpreter for Microcomputers, Part 1"

The following letter from Fred J Dickey contains corrections to "An APL Interpreter for Microcomputers, Part 1" by Mike Wimble, which appeared on page 50 of the August 1977 BYTE. We thank Fred for his efforts.

I received my August 1977 BYTE yesterday, and was quite impressed by Mr Wimble's APL implementation article and the fact that he is giving a *hardware independent* description of a significant software system. This article is of value regardless of what type of microprocessor one uses. Furthermore, one would expect the article to be of value as long as there is interest in APL, which will probably be long after the current crop of microprocessors become historical curiosities.

Despite my enthusiam, I regret to inform you that I found the following errors by doing a hand simulation of the program on page 55.

1. Some arrays are dimensioned starting at 0, others at 1. In particular TVAL starts at 0, and all others seem to start at 1. This is not explained anywhere.
2. FUNC, NOMBRE, OTHERS, NILAD, MONAD, DYAD, EOL, CARRET are called *subroutines*, but they are in fact extensions of the main routine given on page 56.
3. The labels IP, IPGET, IP_GET, and IPINIT are nowhere defined. Apparently IP = IPGET = IP_GET and the flowchart on page 56 should appear as shown in figure 1.
4. Why is DA initialized to 3? Also, the scanner initialization box on page 56 should appear as in figure 1.
5. "No" and "yes" on page 56 should also appear as in figure 1.
6. The call to IDEN on page 56 should say CALL IDEN (Q, B). Otherwise Q and B are undefined.
7. The flowchart on page 62 should appear as in figure 2.
8. On page 64, it should be made clear that F and Q are local parameters of subroutine FN_VAR_ADD. F and Q have different meanings external to this routine.
9. On page 64, "routine" carret references STMT. STMT must be in ROM. What is its value?
10. The flowcharts do a good job of trapping errors. How do you recover?
11. Let "_" mean blank. On the example of page 55, you state that you are going to scan 37¯25, but apparently 3_7¯ 2_5 is scanned instead.
12. On page 55, SP(19). C = 6. I don't see how this can be. Also, SP(9). C = 4 not 6, and SP(2). P = 0.
13. Make the following change on page 64:



Fred J Dickey
3420 Granville Rd
Westerville OH 43081

Mike Wimble replies:

Mr Dickey is correct on most points; however, I would like to clarify the following:

Point 3: *IPINIT is of course the beginning of the statement interpreter as defined in part 2 of the article published in September.*
Point 4: *I inadvertently included DA in this portion of the article. It is used in a later version of the interpreter to handle threaded lists.*
Point 9: *STMT, again, is part of the later version of my interpreter, and should be ignored.*
Point 10: *This version of the interpreter has no provisions for error recovery.*
Point 12: *SP(19).C was incorrectly set equal to 6; it should equal 8. SP(2).P is correct as it stands. Although I did not state it explicitly, the case for SP(I).C=1 is used in the later version of my interpreter. It indicates that P is not to be used at that time (This*

Continued on page 164

Figure 1.

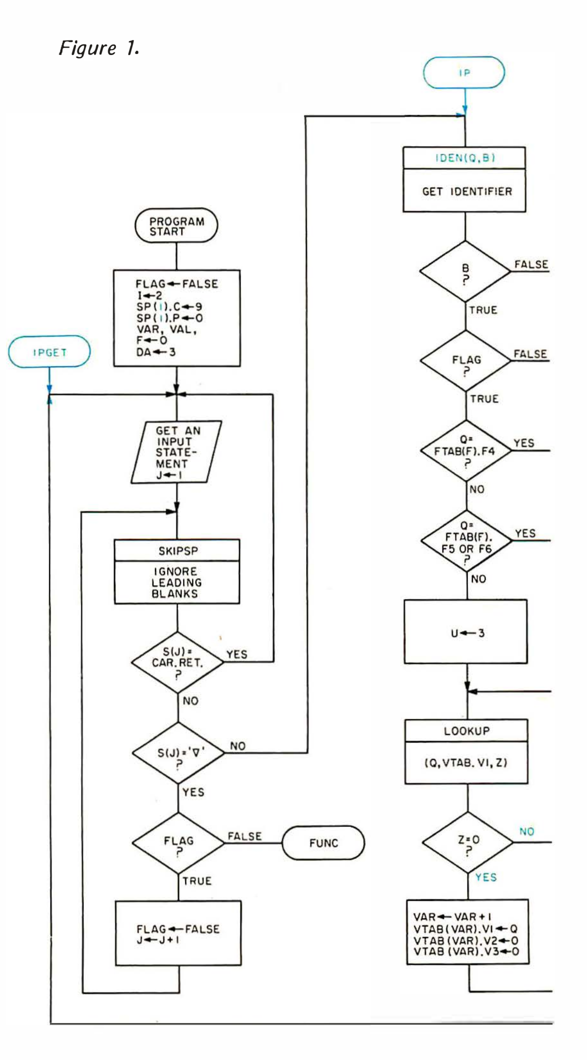
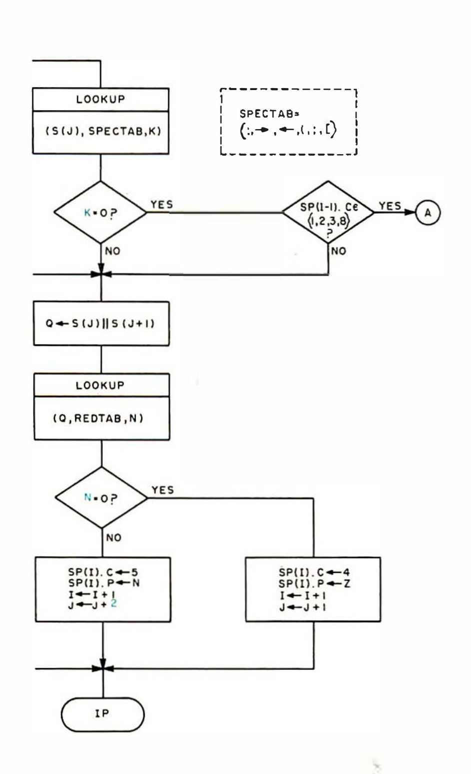


Figure 2.



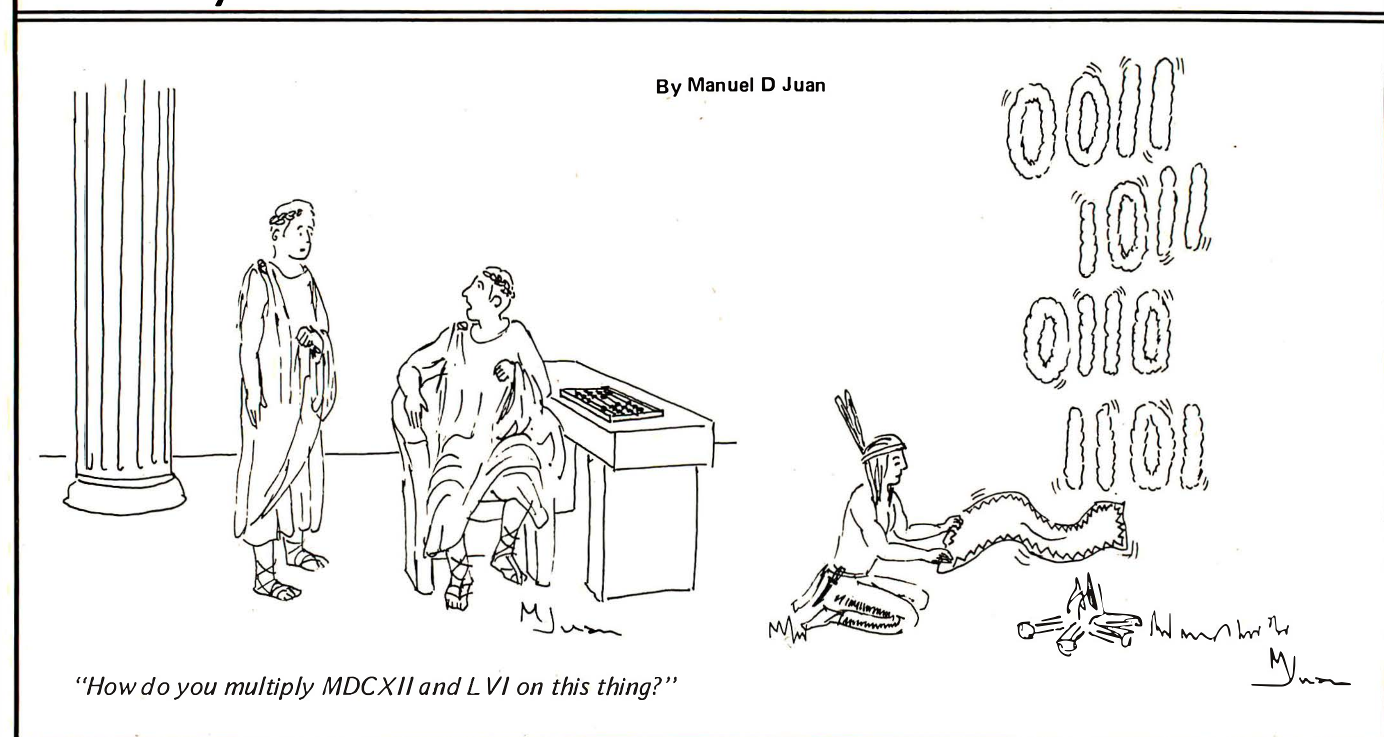
BOMB Lands on APL

Readers of the August 1977 BYTE voted for APL all the way. The BOMB first prize of $100 goes to Mike Wimble for his article, An APL Interpreter for Microcomputers, Part 1, on page 50. The $50 second prize goes to Dr Kenneth Iverson for Understanding APL, page 36. The distribution of points for August's articles was relatively even in the voting (The standard deviation was only 10% of the mean of all article votes.), indicating a diversity of interests on the part of BYTE readers. Mike Wimble's article was 1.7 standard deviations above the mean, and Dr Iverson's article was 1.3 standard deviations above the mean. Readers are encouraged to express their opinions about this month's articles by filling out and sending in the BOMB card between page 256 and the inside back cover.■

Kilobyte™

Readings for Your Next Squantum

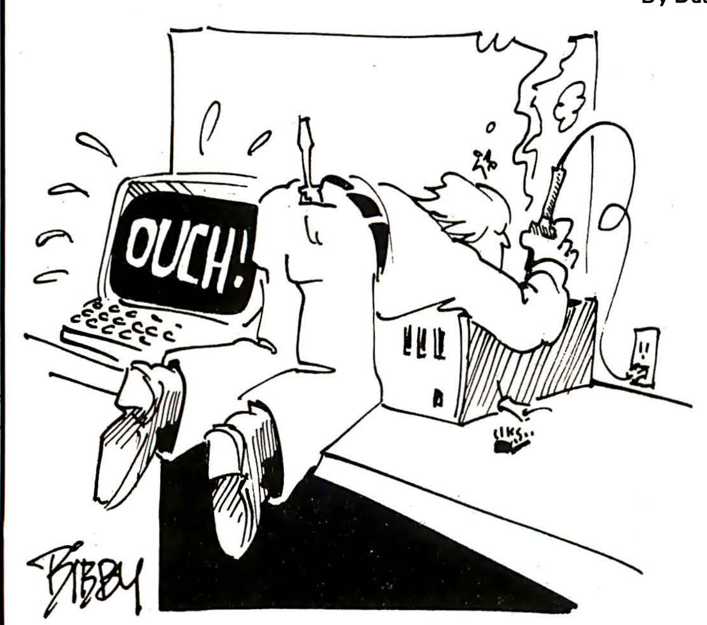
By Manuel D Juan

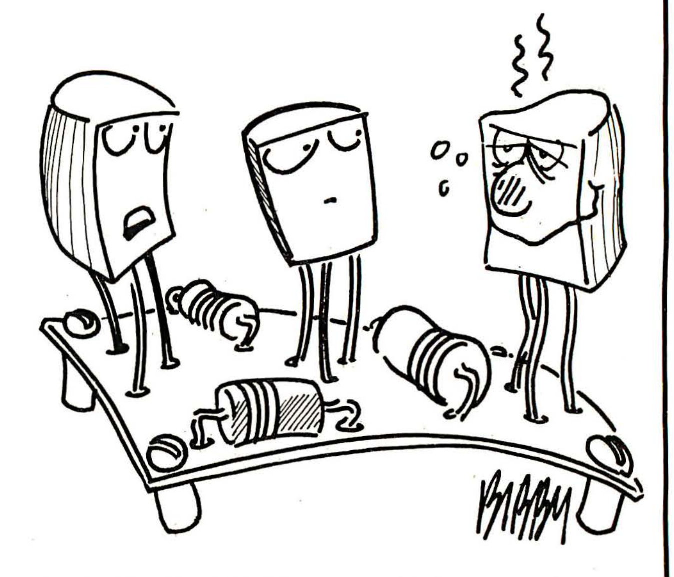


"How do you multiply MDCXII and LVI on this thing?"



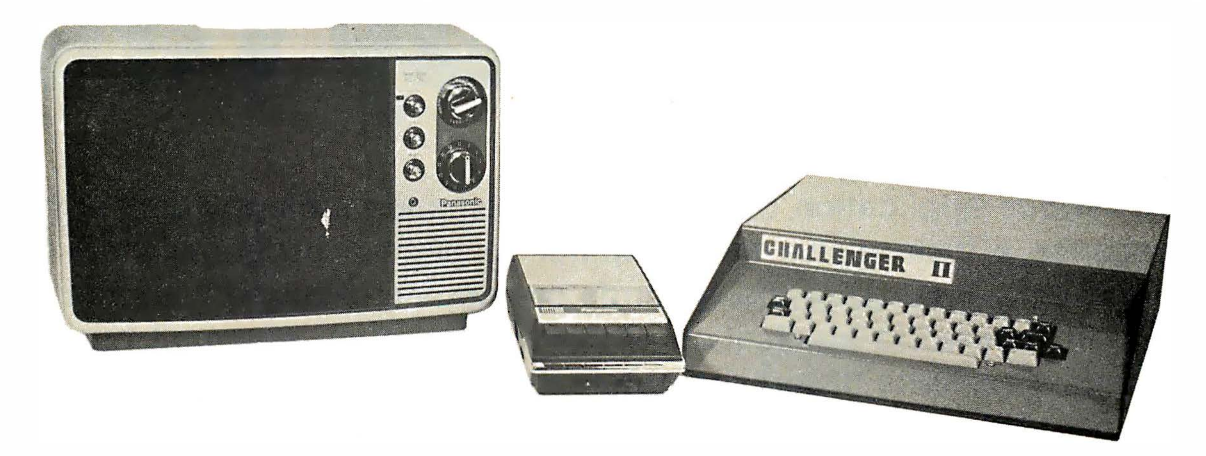
By Duane Bibby





"Maybe I'm biased, but he seems a little saturated to me."

Meet Challenger IIP from Ohio Scientific.



Unlike any other personal computer available today

Complete with BASIC in ROM and 4K RAM, Challenger IIP is the ideal computer for programs in BASIC.

BASIC is there the instant you turn the computer on with a full 32 x 64 character video display. Challenger IIP also comes with an Audio Cassette Interface for program storage. The user simply connects a Video Monitor or a TV via an RF Converter (not supplied) and the machine is ready to use.

Challenger IIP is ideal for both the home user who is new to computing or the experienced user who wants expansion capabilities. Challenger IIP comes with a four slot backplane and is expandable via the full Ohio Scientific product line, which includes 15 system boards offered in over 40 different versions.

Ohio Scientific has always maintained upward expandability from old models to new models, which is nice to know considering the rate at which technology is constantly improving. For example, Ohio Scientific's original 400 series products can be plugged right into the new Challenger IIP. And Ohio Scientific has 2 years of experience in building personal computers, so we're not new to this business unlike some of our competitors.

Complete with a full computer keyboard Challenger IIP comes fully assembled for $598 from Ohio Scientific.

Check the chart below and compare Challenger IIP with other BASIC in ROM computers. Unlike other personal computers, Challenger IIP has a much greater capacity for expansion and the capability to perform big computer functions with all of its big computer features.

	Ohio Scientific Challenger IIP	Other BASIC in ROM Computers
Processor	6502A	6502 or Z-80
Clock	1 or 2 MHz	slower
Display (Lines/Characters)	32/64	25/40 or 16/64
Keyboard	Full Computer (Capacitive Contact)	4 Function Calculator Type or Full Computer (Mechanical Contact)
Display Characters	256	128 or 64
Lower Case	Yes	No
Plotting	Yes	Yes
Audio Cassette Interface	Yes	Yes
BASIC	8K By Microsoft	some have only 4K BASIC
String Functions PEEK, POKE, User	Yes	Not Always
Machine Language Accessible	Yes	Not Always
Optional Assembler/Editor	Yes	No
Disk Option Available Now	Yes	No
In Case Memory Expansion Ability	36K	Less
Expansion Boards Available Now	15	None

Disk Based Co
by Ohio

Any serious application of a computer demands a Floppy disk or hard disk because a disk allows the computer to access programs and data almost instantly instead of the seconds or minutes required with cassette systems. In real-world application of computers, such as small business accounting, a cassette based computer simply takes too long to do the job.

Ohio Scientific offers a full line of disk based computers utilizing full size floppy disks with 250,000 bytes of formatted user work space per disk. That's 3 to 4 times the work space of mini-floppies.



Challenger II

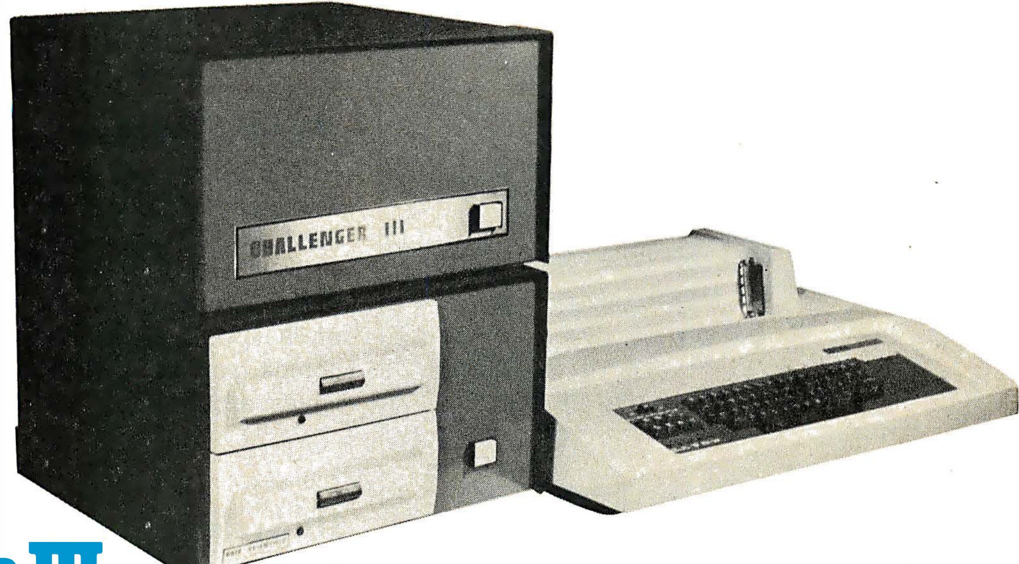
Challenger II is available with a single or dual floppy disk and a minimum of 16K of RAM instead of ROM BASIC. The disk BASIC is automatically loaded into the computer so there is no need for ROMs.

Ohio Scientific's powerful disk operating systems allow the computer to function like a big system with features like random access, sequential, and index sequential files in BASIC and I/O distributors which support multiple terminals and industry-standard line printers.

Challenger II's with disks can have the following optional features:
• 16 to 192K of RAM memory • Single or dual drive floppys • Serial and/or video I/O ports • Up to 4 independent users simultaneously • Two standard line printer options • Optional 74 Megabyte Hard disk • Much more

Challenger II disk systems are very economical. For example a 16K Challenger II computer with serial interface, single drive floppy disk, BASIC and DOS costs only **$1964.00** fully assembled.

mputer Systems Scientific



Challenger III

Ohio Scientific proudly announces the ultimate in small computer systems, the Challenger III. This computer has a 3 processor cpu board equipped with a 6502A, 6800, and Z-80.

This system allows you to run virtually all software published in the small computer magazines!

The Challenger III is fully software and hardware compatible with Ohio Scientific products and can run virtually all software for the 6800, 8080 and Z-80 including Mikbug® dependent 6800 programs!

Incredible as this is, Challenger III costs only about 10% more than conventional single processor microcomputers. For example a 32K Challenger III with a serial interface and a dual drive floppy disk (500,000 bytes of storage) costs only **$3481.00.** Fully Assembled, complete with software. Terminal not included.

☐ **Send me the Fall '77 Catalog. I enclose $1.**
☐ I would like to order directly from this advertisement. (Please allow up to 60 days for delive y)

NAME________________________

ADDRESS________________________

CITY____________ STATE________ ZIP________

To order: Payment by:
BAC (Visa)______ MC______ Money Order______
Credit Card Account #____________
Interbank # (Master Charge)____________

______ Model 500 Boards @ **$298.00** ______

______ Challenger IIP @ **$598.00** ______

______ Super Kit @ **$398.00** ______

______ 16K Challenger II complete with serial interface, single drive floppy disk, BASIC and DOS @ **$1964.00** ______

______ 32K Challenger III with serial interface, a dua! drive floppy disk (500,000 bytes of storage) @ **$3481.00** ______

Ohio Residents add 4% sales tax ______

TOTAL CHARGED OR ENCLOSED ______

Order directly from: Ohio Scientific, 11679 Hayden St., Hiram, Ohio 44234 or your local OSI dealer

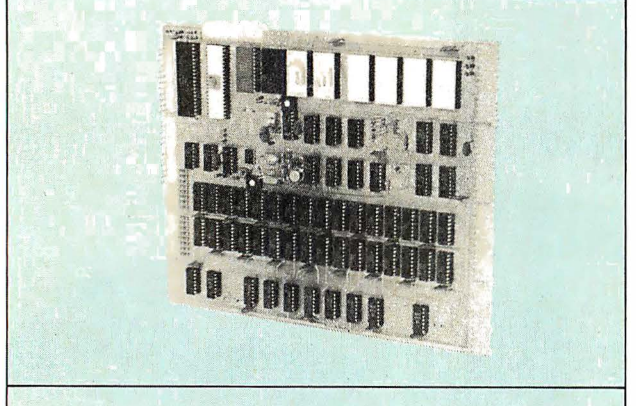
All orders shipped insured UPS unless otherwise requested.

OHIO SCIENTIFIC

11679 Hayden • Hiram, Ohio 44234

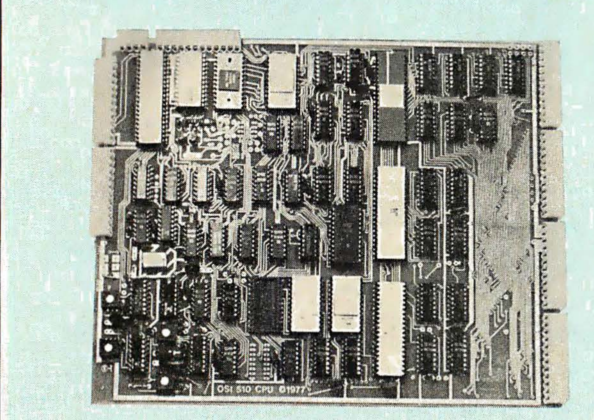
Introducing three boards only Ohio Scientific could build.

Ohio Scientific provides 15 system boards offered in over 40 different versions for Ohio Scientific Computer users. All of the boards are compatible with Ohio Scientific systems and many of them are by far technologically superior to any other microcomputer products on the market. And Ohio Scientific has the technology that made them possible.



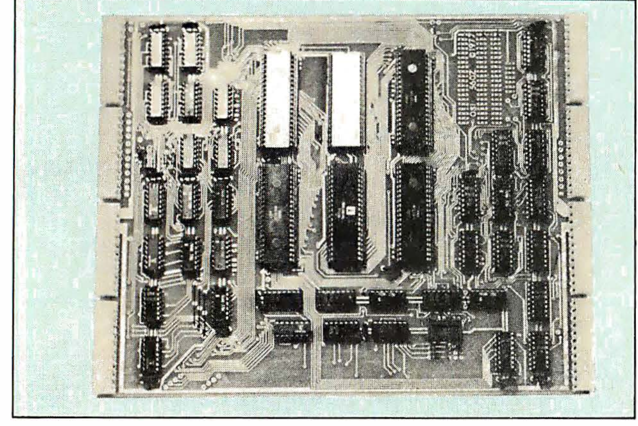
500 CPU Board

This board gives you our ultra-fast 8K BASIC in ROM with plenty of user workspace (4K RAM) for as little as $298.00. Use it as a standalone or as the CPU in a large system. BASIC is there the instant you turn it on. And in the October issue of *Kilobaud Magazine,* our version of 8K BASIC came out the winner in a BASIC timing comparison test of all of our competitors. The 500 is the fastest around!



510 Systems CPU Board

This is our unbelievable triple processor board! Complete with the 6502A, 6800, and Z-80 processors, this board allows you to run virtually all programs published for small computers. Available in the Challenger III, the 510 board is ideal for industrial development and research applications. There isn't another triple processor board like the 510 anywhere, except at Ohio Scientific!



560Z CPU Expander Board

The 560Z board is our multiprocessing board with a Z-80 and 6100 chip. This board allows you to run several processors simultaneously and the 6100 chip lets you run powerful PDP8 software with the 560Z. The 560Z board is the only multiprocessing board available for small computers, and Ohio Scientific makes it!

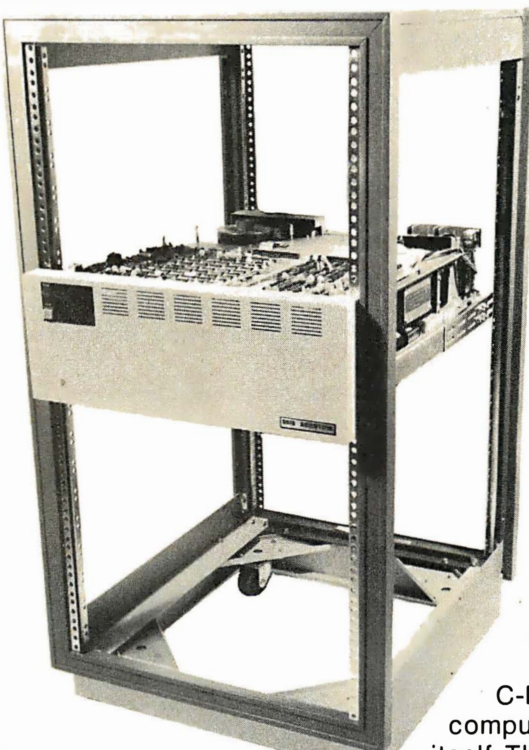
These three state-of-the-art CPUs are only a small part of the picture. Ohio Scientific's advanced technology offers you other unique features such as Multiport Memories, Distributed Processing, Big Disks with up to 300 megabytes on line, and Advanced Software.



11679 Hayden • Hiram, Ohio 44234

Announcing the most advanced disk anywhere for $6,000

The 74 megabyte disk from Ohio Scientific



C-D74 from Ohio Scientific is the ultimate storage device for small computers.

The C-D74 is the first Winchester technology disk for small computers making big system technology affordable and reliable for the small system not under maintenance contract.

The disk uses a non-removable sealed chamber drive with a unique rotary positioner to provide the highest performance disk available today.

The Ohio Scientific C-D74 can store all the records of a medium size company for instant access. And the Winchester technology of the C-D74 means that the drive can run 24 hours a day without worry of disk wear.

There are other important C-D74 applications in business computing and research in computing itself. The disk makes small computers practical for much larger jobs than formerly thought feasible, particularly since most business computing is disk bound and not computer bound.

C-D74 provides an unbelievable 35 millisecond average access time to any of 74 million bytes of information. With a 10 millisecond single track seek, the drive has an incredible data transfer rate of 7.3 megabits per second.

Recommended minimum hardware for the C-D74 is a Challenger with 32K RAM and at least 8K on a Dual Port 525 board, and a single or dual-drive floppy disk.

The drive , cable, interface for an Ohio Scientific Challenger and OS-74 operating system software is $6,000 FOB Hiram, OH. Equipment rack shown not included.

OHIO SCIENTIFIC DEALERS

Abacuz Stores Limited
55 Erb St. East
Waterloo, Ontario
Canada N2C 3EO
(519) 885-1211

American Microprocessors Equipment & Supply Corp.
20 N. Milwaukee Ave.
Prarieview, IL 60069
(312) 634-0076

Century 23
4566 Spring Mountain Rd.
Las Vegas, NV 89102

Computer Mart of New York
118 Madison Ave.
New York, NY 10010
(212) 686-7923

Computer Power
P.O. Box 28193
San Diego, CA 92128
(712) 746-0064

Delaware Microsystems
92 E. Main St. #1
Newark, DE 19711
(302) 738-3700

Desert Data
Microcomputer Sales
P.O. Box 1334
Tucson, AZ 85702
(602) 623-6502

The Home Computer Co.
P.O. Box 1891
University Station
Charlottesville, VA 22903
(804) 295-1975

Microcomp
P.O. Box 1221
Fond-Du-Lac, WI 54935
(414) 921-4669

Microcomputer Workshop
234 Tennyson Terr.
Williamsville, NY 14221
(716) 634-6844

Omaha Computer Store
4540 S. 84th St.
Omaha, NE 68127
(402) 592-3590

REPRESENTATIVES

Abacuz Data
P.O. Box 276
Oil City, PA 16301

Associates Consultants
33 Ogden Ave.
East Williston, NY 11596
(516) 746-1079

BRAG Microcomputers
19 Cambridge St.
Rochester, NY 14607
(716) 442-5861

Computer Business
P.O. Box 171
LaPorte, IN 46350
(219) 362-5812

Johnson Computer
P.O. Box 523
Medina, OH 44256
(216) 725-4560

Omega Computing, Ltd.
Box 220, Station P
Toronto, Ont. M5S 2S7
(416) 424-2174

Pan Atlantic Computer Systems, GmbH
61 Darmstadt
Frankfurterstrasse 78
West Germany
(08102) 3206

Science Education Extension
11516 LeHavre Dr.
Potomac, MD 20854
(301) 299-9506

Spectrum Technology Services
P.O. Box 942
Palos Verdes Estates, CA 90274

Tek-Aids, Inc.
1513 Crain St.
Evanston, IL 60202
(312) 328-0110

The state of the art in small computers.

To order direct call 1-216-569-3241

The TRS-80: Radio Shack's New Entry into the Personal Computer Market



Photo 1: The New Radio Shack TRS-80 home computer system. Shown are the keyboard, video display monitor, instruction manual and prototypes of the upcoming memory expansion module and disk drive.

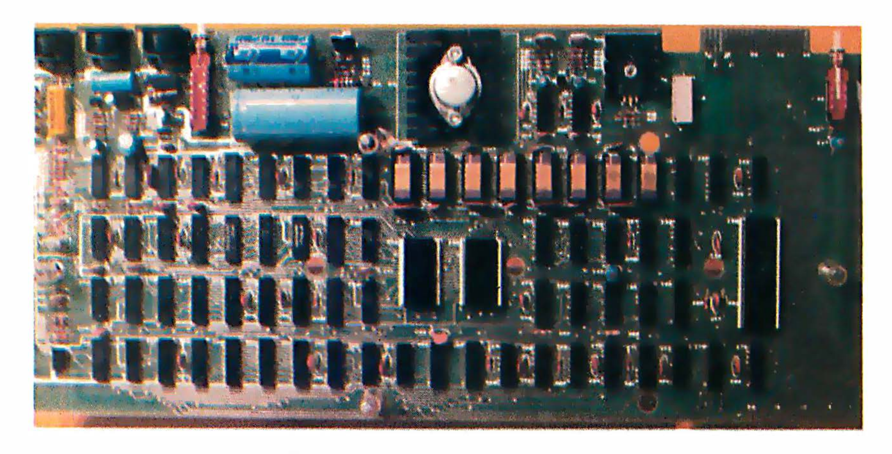


Photo 2: The single board Z-80 processor which forms the heart of the TRS-80. Note the 40 pin IO connector at upper right.

Text and Photos by
Chris Morgan, Editor

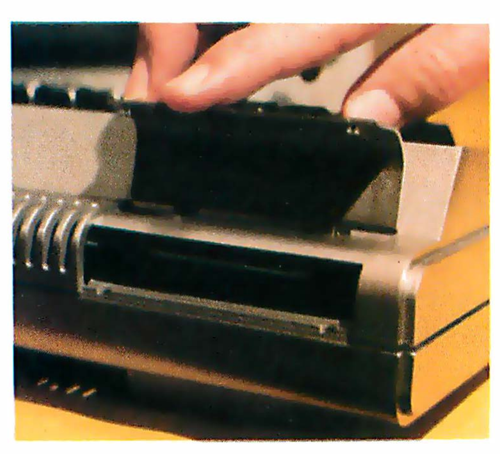


Photo 3: Rear view of the Radio Shack computer showing the 40 pin IO connector.

Announced in August, the new Radio Shack TRS-80 is a major entry into the personal computer market. The $599 single board Z-80 based unit comes complete with a full ASCII character set keyboard, cassette recorder and video display monitor. Also included for the price is 4 K bytes of programmable memory and 4 K bytes of read only memory; the latter features a built-in BASIC package. An additional 12 K bytes of programmable memory can be added for $289.

The computer is being marketed in selected Radio Shack stores across the country; peripherals planned for release in December include a disk drive, printer and memory expansion hardware. An interesting feature of the TRS-80 is the convenient hinged door on back for easy access to the 40 pin printed circuit card IO connector.

Software will be available in a variety of packages, including a blackjack program (which comes free with the computer); a payroll program for up to 15 people, priced at $19.95; a kitchen menu program for $4.95; and so on.

The unit is priced competitively with some other computers on the market, and it will be interesting to see what develops in this low priced appliance computer market.■



Photo 4: A closeup of the forthcoming microcomputer expansion module and disk drive.

You don't have to be a million dollar corporation to profit by distributing this Data General micro-computer system with our BASIC business application packages for:

• Invoicing with Inventory Control
• Accounts Receivable • Sales Analysis
• Accounts Payable • Payroll • General Ledger



We are seeking data processing oriented individuals and companies to sell and install these systems in their areas. Liberal distributor discounts start at quantity one. Hardware installation and maintenance is available directly from your local Data General office.

For more information, use the coupon.



Mini-Computer Business Applications
4929 Wilshire Blvd., #940
Los Angeles, CA 90010
Phone: (213) 936-7131

NOVA® is a registered trademark of Data General Corp.

Please send me information about distributing MCBA/Data General systems.

Name ______________________

Company ______________________

Address ______________________

City ______________________ St ________

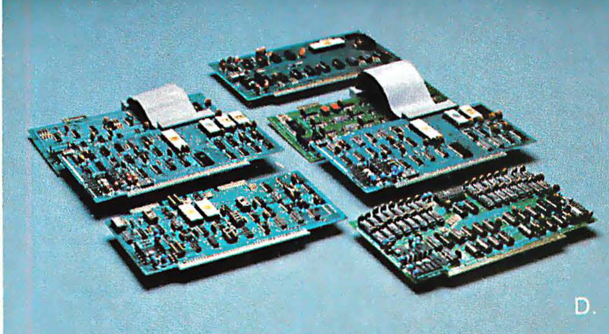
Zip ____________ Phone ______________________

BY1

A.

B.

C.


D.



E.

A. VDP-80 Computer with 300 lpm printer.

B. PCS-80 with CRT, dual floppy disk & Intelligent Keyboard options.

C. Peripherals—(clockwise from left) 45 cps daisy-wheel printer/terminal, 24x80 CRT terminal, 45 cps daisy-wheel printer, Intelligent Breadboard, 44 col. alphanumeric line printer.

D. Processor, Memory & Interface boards—shown MPU-A, 65K RAM, and floppy disk, line printer and serial I/O's.

E. PCS-80 System—sample component configurations.

Microcomputers: Just Ask IMSAI.

If you wonder who leads the way in technology, look into IMSAI's list of industry firsts—IMSAI 8048, first complete control computer on a board; IMSAI 65K RAM Board, first to offer four times the memory capacity previously available on one board; IMSAI printers, first with high-speed direct memory access.

If you wonder why IMSAI products have gained the reputation for the standard of excellence in microcomputer systems, check with any one of the more than 10,000 IMSAI owners.

If you wonder who offers the broadest line of hardware, software, and peripherals, visit any one of the more than 275 IMSAI dealers around the world.

If you wonder how microcomputing can fit your specific needs, ask IMSAI. Because when it comes to microcomputers, we have the answers.

An IMSAI Product to Answer Every Microcomputing Need:

Let's start with our product line. In all, IMSAI offers more than 120 high quality, completely integrated systems, components, peripherals and software. Here's just a sampling:

Single Board Central Processors:
- MPU-A (8080 based)—Industry standard.
- MPU-B (8085 based)—50% faster 8080.
- 8048—Programmable control computer.

Interfaces:
- Video I/O—24x80 CRT. Edit & data entry.
- Serial I/O—2 port I/O, all std. protocols.
- Parallel I/O—4 & 6 port TTL level I/O.
- Multiple I/O—2 cassette, 2 parallel, 1 serial & 1 control I/O.
- DMA—For floppies & line printers.

Peripherals:
- Printers—40/80/132 col. 30 cps-300 lpm.
- Video displays—Large assortment.
- Tape Drive—9 track, 800 bpi, 25 ips.
- Floppy Disks—Single/double density.

Memory Expansion Boards:
- 4K RAM—Programmable memory protect.
- 16/32/65K RAM-16K paging option for virtual memory addressing.
- Intelligent Memory Mngr.—Handles up to 1 megabyte.

Self-Contained Systems:
- VDP-80—Computer/terminal/mass storage unit. Assembled & tested.
- PCS-80—Integrated component system.

Software:
- DOS—Enhanced CP/M.
- BASIC—Interactive or compiler with scientific and/or commercial features.
- FORTRAN IV—Level 2 ANSI compiler.
- Self-contained Systems:
  SCS 1 & 2/TCOS—Assembler/line editor/debugger.
  4 & 8K BASIC—Optional cassette support.

Compare IMSAI. You'll realize that ours is the most complete product line available. Whatever your needs, you can get them from one source. IMSAI.

A wide selection of components is only the beginning. IMSAI offers much more. Just ask.

Answers For Businessmen:

Announcing IMSAI's VDP-80. This totally self-contained unit includes a megabyte of disk memory via floppy disk, 32K computer memory (expandable to 256K), 12" CRT and 62 pad main keyboard with 10 pad numeric keyboard. Several printer options available.

If you want speed and accuracy in high volume work such as word processing, or business data collection and analysis, the VDP-80 is your cost effective answer.

Answers For The Personal User & Educators:

Introducing IMSAI's new PCS-80 System, the fully integrated microcomputer component system, configurable to your exact needs. The basic system consists of our Intelligent Keyboard and the PCS-80 which houses an 8085 based CPU, 16K of RAM, intelligent ROM monitor, serial I/O port, 24x80 CRT, with an extra 7 slots in the chassis for expansion.

System component options include single or dual mini and standard floppy disks. The choice is yours, configure the system as you like.

IMSAI has answers for the educator, too. Take the basic PCS-80, add 8K of PROM, 4K of RAM and our self-contained 8K BASIC software, and you have a complete operating system your entire department can use to teach anything from elementary programming to advanced computer science.

Require a bit less sophistication? Use our Intelligent Breadboard system for learning, designing and building microcomputer assemblies.

Rather do it from scratch? Start with our single board MPU-B central processor, the heart of the PCS-80 System. It has a 1K ROM monitor, 256 bytes of RAM and serial and parallel I/O.

Since the MPU-B is 8085 based, you can run all programs previously developed for the 8080, 50% faster. Without requiring faster memory.

Answers For Industry:

IMSAI products provide the expandability and flexibility manufacturers demand for microcomputing applications.

We offer rack mountable components for the standard 19" RETMA racks, powerful MPU boards, I/O and memory boards for easy system expansion and configuration, and a broad line of peripherals and subsystems fully integrated and ready to go to work.

IMSAI has what you need to make tomorrow's design today's reality.

Answers For Current IMSAI Users:

There are over 10,000 of you. And, we haven't forgotten. You might say that we thought of you before you even thought of us.

That's why every new product is designed to accommodate expansion, rather than outdate equipment.

For example, our new PCS-80 retrofit kit comes complete with MPU-B, replacement front panel photomask and additional hardware bracketing. So you can enjoy a single cabinet PCS-80 computer, with your choice of integrated component configurations.

The Answer For Everyone:

Dial (415) 483-2093, Ext. ACT. That's IMSAI's action hotline. Designed to answer the thousands of questions we didn't have a chance to answer in the space of this ad.

Call us. We'll assist you in putting together a system, direct you to your nearest IMSAI dealer, and send you our new catalog with all the details.

In short, if you have any questions at all regarding microcomputers, put us to the test.

Just ask IMSAI.

IMSAI®

The Standard of Excellence in Microcomputer Systems.

IMSAI Manufacturing Corporation
14860 Wicks Blvd., Dept. B-9
San Leandro, CA 94577
(415) 483-2093 TWX 910-366-7287

Features and specifications subject to change without notice.

Circle 67 on inquiry card.

M F Smith
Research Assistant
Department of Oceanography
University College
Galway IRELAND

Using Interrupts for Real Time Clocks

We have developed several software timekeeping routines for oceanographic data systems which may be of more general interest. These routines are based upon the Motorola M6800 and have been tested on SwTPC 6800, MITS 680b and Motorola MEK-6800D1 evaluation kit systems. The routines require little memory or hardware and do not slow program execution appreciably. Features of the routines are:

- packed BCD storage of time values: days, hours, minutes and seconds.
- little interference with user routines through use of interrupts.
- usable with a wide range of clock frequencies.
- minimal hardware complexity.
- possibility of event scheduling.

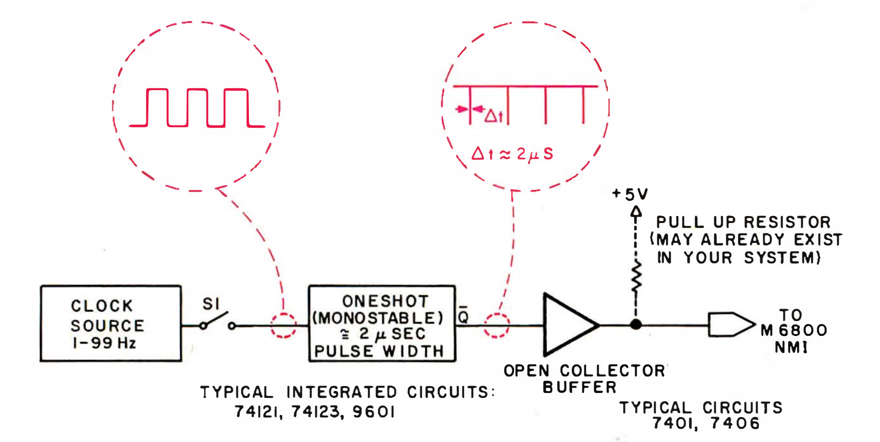


Figure 1: The hardware configuration required for a real time clock implemented with an interrupt line. For the 6800 processor, the negative going pulse of the monostable (oneshot) should be at least two processor cycles in length. The switch S1, or its logic circuitry equivalent, is essential in order to disable the interrupts if user programmable volatile memory contains the interrupt routines. If this switch or its equivalent is not present, receiving interrupts from NMI in absence of an interrupt routine (following power on) leads to quite unpredictable results in the behavior of the system.

Hardware

The routines are driven by direct non-maskable interrupts of the processor by a clock pulse source as shown by figure 1. Use of the NMI in this fashion precludes use for other functions but minimizes hardware. Also, such use of interrupts can cause problems when timing loop software is interrupted: constants which are valid without interrupts can be incorrect when interrupts are in operation. With these caveats in mind, however, use of interrupts proves quite convenient.

The clock source may be in the range 1 to 99 Hz (10 Hz is used here) and drives a monostable (74121, 9601, etc). The Motorola literature describing the 6800's non-maskable interrupt function is just a trifle confusing. Using the information in the *M6800 Microprocessor Applications Manual*, one could conclude that the NMI line requires a low *level* input to initiate an interrupt. This conclusion results from the terse description of NMI and reference to the fact that NMI is supposed to work similar to IRQ. However, the hardware specification sheets for the processor explicitly state that NMI is sensitive to the negative going edge of the digital signal on its input. This detail is easily confirmed by experiment. *[It is also the only sensible way to handle this interrupt, in view of the fact that it cannot be masked in the processor to inhibit further interrupt while the interrupt routine is in operation . . .CH]* The oneshot in figure 1 should be interpreted as a way of transforming an arbitrary signal into a well-defined TTL pulse of a minimum 2 microseconds in length, or slightly greater, which provides the required negative edge.

Unless the time routine is stored in ROM with "hard" NMI vectors, means of disabling NMI pulses must also be provided until the interrupt routine and vector are estab-

SANTA and SCELBI HAVE A UNIQUE GIFT IDEA FOR EVERYONE ON YOUR LIST!

Scelbi Software Books

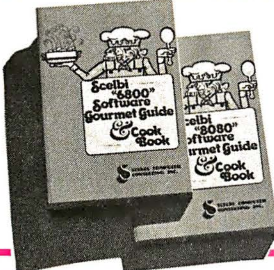
ORDER NOW! SAVE! BEFORE 1978 PRICES GO UP!

FOR THE BEGINNER . . .



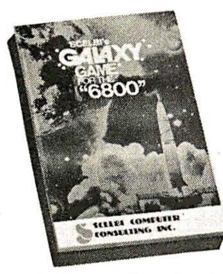
Understanding MicroComputers and Small Computer Systems. A profusely illustrated, easy-reading "must" book explaining fundamental concepts behind operation of microcomputers. Simple English. Gives extra knowledge to read and understand computer magazines and manufacturers' literature. Makes you feel "at home" around computers. Accepted as the standard for the neophyte, you must own this 300-page no-nonsense, easy-reading text. Includes easy-to-use glossary of key microcomputer oriented words. Order now. Save! **$9.95 each ppd.**

FOR THE INTERMEDIATE . . .

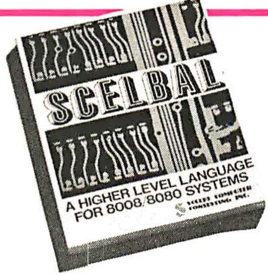


Scelbi's Software Gourmet Guides and Cookbooks for '8080' or '6800' lets you cook up mouthwatering programs. Delectable "how to" facts, including '8080' or '6800' instruction sets. How to manipulate stacks. Flow charts. Source listings. General purpose routines for multiple precision operation. Programming time delays for real time. And lots more. Even floating point arithmetic routines! Order your copies today. Bon appetite! Specify: '8080' or '6800'. **$9.95 each ppd.**

FOR THE ADVANCED . . .



GALAXY Microcomputer Outer Space War Games for '6800'. Captain your own starship on intergallactic journeys filled with battles, refueling problems, weaponry, warp factors, and more — all against your '6800'. A complete book, written in machine language for 4K memory. Ever-changing interstellar adventure, includes source listings, flow charts, routines, more. Order today. Blast off tomorrow! **$9.95 ppd.**



SCELBAL. Higher Level Language for '8008'/'8080' Systems. Complete, illustrated program book. Routines. Techniques. Source listings. Flow charts. More. Includes 5 commands, 14 statements, 7 functions, and it runs in 8K and more. All you need to customize a high level language for your system at a fraction of the cost. Order today! **$49 ppd.**



The '8080' Programmer's Pocket Guide; '8080' Octal Code Card and/or '8080' Hexadecimal Code Card. Compact pocket guide for instant reference to either code card. Cards are instant slide rule aids for programming/debugging '8080' software. Standard mnemonics with corresponding codes. Color coded instructions indicate which flags are affected during instruction execution. Quick, logical reference formats. ASCII code chart for 128 characters. '8080' status words. Register pair codes. More. Order all three now . . . **only $2.95 per item.**

Prices shown for North American customers. Master Charge. Postal and bank Money Orders preferred. Personal checks delay shipping up to 4 weeks. Pricing, specifications, availability subject to change without notice.



SCELBI COMPUTER CONSULTING INC.
Post Office Box 133 PP STN
Milford, CT 06460 Dept. B

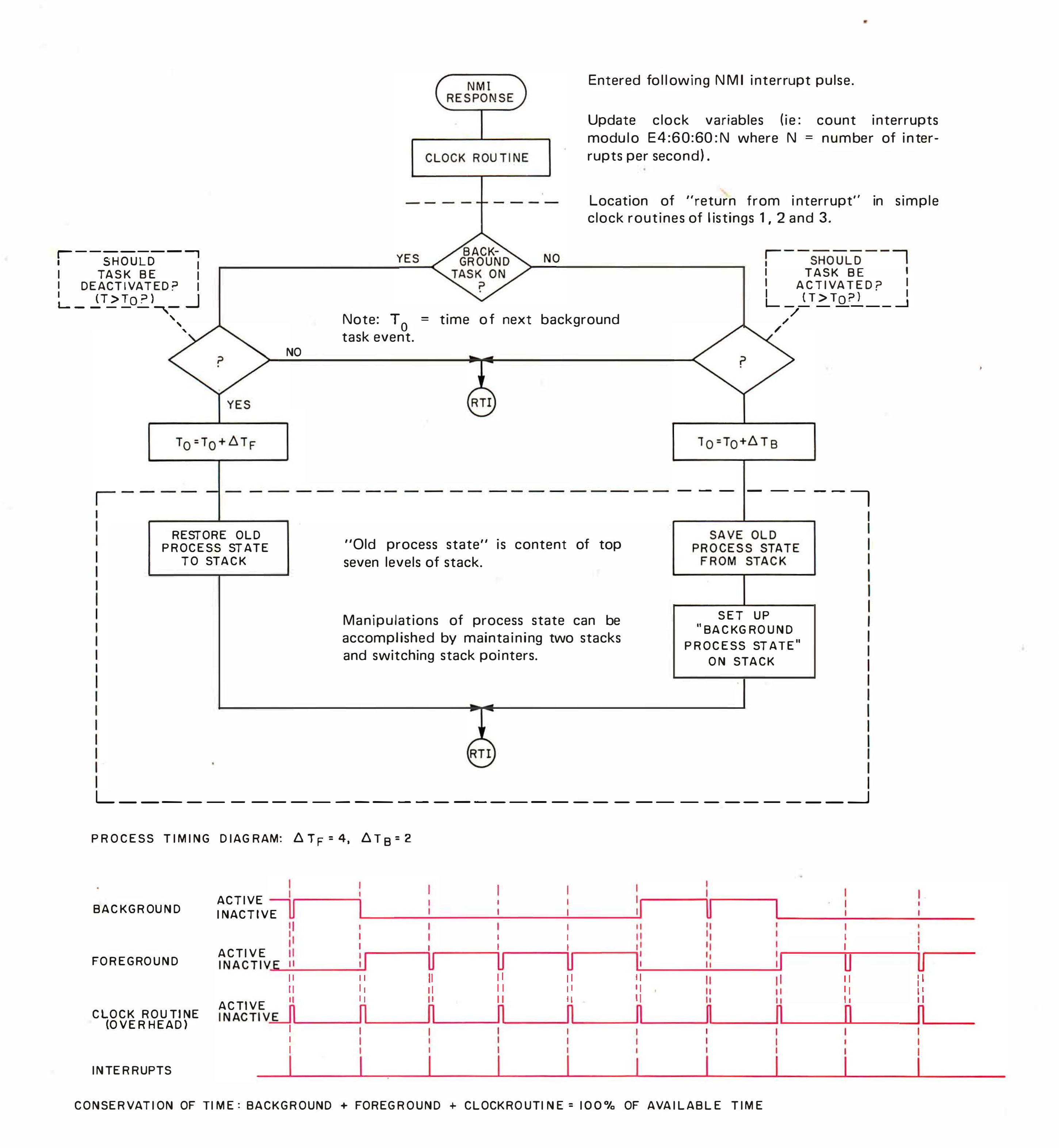


Figure 2: A suggested algorithm for implementing two simultaneous tasks using the interrupt input to keep track of times ΔT_B and ΔT_F allocated to each process. It is assumed here that the "foreground" task is the principle task, and that the presence or absence of a hidden "background" task is governed by a flag.

lished in programmable memory. We use a mechanical switch (S1), but more elegant methods are possible with increased hardware complexity.

Software

A minimal timekeeping routine called RAMTIME is shown as listing 1. This routine performs the function of a real time clock when it responds to the interrupts from NMI. It has two counters. A counter 1 byte long called WATCH continually cycles with a binary integer count. A second 5 byte count field provides the usual day, hour, minutes and seconds counts using the "overflow" constants 99, 99, 24, 60, 60 and the number of interrupts per second to determine when a carry has occurred. All the counting in this field is done in BCD. If at any time it is desired to output the BCD numbers in the various count fields, the MIKBUG subroutines OUT2HS and OUT4HS can be used to convert to external ASCII decimal values on a terminal.

The program includes a binary "stopwatch" function. The location WATCH is incremented with every NMI pulse, thus providing a convenient means of timing short events. This function can be eliminated with a small saving of memory, if desired.

Clock rates different from the 10 Hz rate are accommodated by changing the RATE variable (RAMTIME) to the packed BCD value of the clock rate, eg: the present rate of hexadecimal 10 (BCD for 10 Hz) is changed to hexadecimal 60 for a 60 Hz clock source.

Scheduling

The nature of the NMI-driven clocks make them ideal for the inclusion of task scheduling routines. Scheduling, using these routines as vehicles, is transparent to the user program, ie: scheduling is performed without "knowledge" of the program that scheduling is going on. Timetables are accurate because the schedule is checked every NMI. A very simple scheduler is suggested in the flowchart of figure 2. This algorithm implements a timing diagram (like that in the figure) which switches between two tasks arbitrarily called "foreground" and "background." This is the simplest form of "timesharing" or "multiprogramming."

```
PAGE  001   RAMTIME

                        NAM    RAMTIME
A04A                    ORG    $A04A
A04A 00                 FCB    0          DUMMY LOCATION
A04B 00         DAY     FCB    0          TIME IN PACKED BCD FORMAT
A04C 00         HOUR    FCB    0
A04D 00         MIN     FCB    0
A04E 00         SEC     FCB    0
A04F 00         SEC1    FCB    0
A050 00                 FCB    0          DUMMY LOCATION
A051 00         WATCH   FCB    0          BINARY 'STOPWATCH' LOCATION
                *
A052 99                 FCB    $99,$99,$24,$60,$60
A053 99
A054 24
A055 60
A056 60
                *
A057 10         RATE    FCB    $10        *CLOCK RATE
                *
                *FOR DIFFERENT CLOCK RATES, CHANCE RATE
                *E.G., FOR 60 HZ CLOCK CHANGE TO $60
                *1-99 HZ ALLOWABLE CLOCK RATES
                *
A058 CE A051    TIME    LDX    #WATCH     TIME PROGRAM BEGINS HERE
A05B 6C 00              INC    0,X        INCREMENT THE STOPWATCH
A05D 09                 DEX               DECREMENT TIME ADDRESSES
A05E 6F 00      DINC    CLR    0,X        CLEAR ON CARRY
A060 09                 DEX               NEXT ADDRESS
A061 86 01              LDA A  #1         DECIMAL INCREMENTATION/CARRY
A063 AB 00              ADD A  0,X
A065 19                 DAA               HALF CARRY
A066 A7 00              STA A  0,X        COMPLETE DECIMAL INC
A068 A1 07              CMP A  7,X        CARRY?
A06A 27 F2              BEQ    DINC       YES, CARRY
                *
                ***SCHEDULER(S) INSERTED HERE***
                *
A06C 3B                 RTI               RETURN TO PROGRAM
                *
                        END

TOTAL ERRORS 00000
```

Listing 1: RAMTIME. This routine is a minimum "clock" and "stopwatch" function to be used at interrupt service of an NMI (nominally 10 Hz rate). The "stopwatch" maintained at hexadecimal location A051 is incremented as a binary number every interrupt for short term timing by counts. After incrementing stopwatch, the routine treats the bytes at locations A04A to A04F as a 12 digit BCD field with subfields for days (2 bytes), hours (1 byte), minutes (1 byte) and seconds (1 byte) and parts of a second (1 byte). The overflow values for each field are coded as BCD numbers stored at locations A052 to A057.

Operation

Startup of the routines is not automatic if routines and vectors are held in programmable memory. The source of NMI pulses must be disabled until the routine and vector are loaded. Once they have been installed, enable the NMI source and the routine begins working. Time can be set using memory alter functions or with special setting routines. Once the timekeeper is running, normal operation may proceed as usual, subject again to the caveat of checking the effects of interrupts on any timing loops in other programs.■

Spikes: Pesky Voltage Transients and How to Minimize Their Effects

John McCain
3523 Hardy St
Shreveport LA 71109

You're sitting at your computer playing a game of Super Universe War, about to defeat King Computer, when suddenly, instead of his spaceship disappearing from the display, you see smoke rings drifting from the top of your mainframe. While you curse the expert technician that built the system (you), you dissect the power supply and find a shorted rectifier diode or a bad regulator integrated circuit. Although the uninformed would blame the component manufacturer, you know that it was Spike that did you in; possibly the voltage spike your brother made when he started the washing machine. The roughest environment you can put that fragile MOS circuit in is probably the one you find most comfortable, your house. The way voltage transients run around the power wiring in your home, you'd think they made the mortgage payment. Let's look at just what these beasts are, where they come from, what they do, and how to protect your microcomputer from them.

The beast I'm talking about is the voltage impulse that enters your computer through the wall plug and tries to eat power supply components and fragile chips. These spikes originate everywhere. You can't turn on the television or turn off the coffee pot without making one. Many are small enough to pass by unnoticed, but often they dump their energy where you least want it. Voltage spikes of 1700 V have been recorded on the 120 V wiring in common houses. Multiple spikes of over 1200 V can be expected in 2 to 4% of all houses. These are usually due to changes in an electrical circuit, ie: opening or closing a switch. Remember, the wiring in your house obeys the same laws of nature that govern other circuits with resistance, inductance and capacitance. If you try to rapidly change the current through an inductor, for example, opening or closing a switch, the voltage across it rises rapidly. Guess what? Most power wiring just happens to be predominately inductive. Researchers have shown that residential areas often exhibit more transients, and more severe transients, than commercial and even some industrial areas. What does the spike look like on an oscilloscope? It is usually a damped sine wave such as the one in figure 1. It has extremely sharp rise characteristics (steep leading edge) and it normally dies out after 5 or 10 cycles. It may be only 5 µs long, but may last for 50 µs or longer. A typical wave shape is shown in figure 1. Another source of surges that I will quickly mention is lightning. Although we can't prevent it, we can divert it. I have a lightning arrester at the power entrance to my house. If you don't, I strongly suggest that you look into getting one. It's a good insurance policy for about $10. I've never seen an electric utility that didn't install lightning arresters like they were going out of style, and those people know what they are doing.

Now, let's look at what a well-placed spike can do. It might find a low impedance

Figure 1: Typical shape of a voltage transient waveform. The voltage transient is superimposed on the normal voltage in a circuit, and is characterized by an exponentially damped envelope around an oscillatory waveform.

The Dumb Terminal lets you put it all together.

With the new, lower-priced Dumb Terminal™ Kit, that is. Pick one up and escape, once and for all, the headaches of scavenged teletypes and jury-rigged TV sets. With just a little time and aptitude, you can have a live and working Dumb Terminal right in your own home, garage, or business. One that lets you get it all out of your system — or into it.

Forget the cheap imitations, with their overblown price tags and interminable lists of options. With the Kit, you can build yourself the same, old basic Dumb Terminal that's been selling over 1500 units a month. With basic, sensible features like a bright 12" diagonal screen. Fifty-nine data entry keys. 1920 characters displayed in 24 rows of 80 letters. Plus 33 positive action switches that let you activate functions like 1 of 11 different baud rates, an RS232C interface, or a 20mA current-loop. And more. Not bad for Dumb.

All you need, besides the Kit, is some initiative, and a few basic tools — a good soldering iron, wire cutters, needle-nose pliers, and one or two trusty screwdrivers. The Dumb Terminal Kit provides you with everything else. Including an attractive cabinet, CRT screen, keyboard, PC board, and all essential electronic components. Naturally, you also get illustrated, step-by-step assembly instructions, not to mention an easy-to-understand operator's manual.

So, if you'd like more input on the Dumb Terminal Kit, just fill out the coupon and we'll send you complete, free information.

Oh, and by the way, just by sending in the coupon, you will be made a charter member of the Dumb Terminal Fan Club. A select organization that will send you your own nifty Dumb Terminal Fan Club Kit, containing: an official certificate of membership; an autographed photo of the Dumb Terminal himself; and a bona fide membership card to prove irrefutably you're "One of Us." (Sorry, limit one kit per person.)

And, if you include a trifling $6.00, you can have your very own Dumb Terminal T-shirt. (No limit at all on these.)

Simply mail the coupon and get the whole assortment. And find out why members of the Dumb Terminal Fan Club are some of the smartest people around.



Dumb Terminal. Fun Club.

Yes, I would like more information about the incredible Dumb Terminal Kit. And don't forget to enroll me as a Charter Member in the Dumb Terminal Fan Club.

Name________ Title(?)________

Company________

Street________

City________ State____ Zip____

Enclosed is $______ for an official Dumb Terminal T-shirt(s).

Please make all checks and money orders payable to LeAnce & Reiser, OK?

Quantity and size(s) of shirt(s) required: ___S ___M ___L ___XL

Rush this application to: Dumb Terminal Fan Club Headquarters c/o LeAnce & Reiser, P.O. Box 17123, Irvine, CA 92714.

"Dumb Terminal" is a trademark of Lear Siegler, Inc., E.I.D./Data Products, 714 N. Brookhurst St., Anaheim, CA 92803, (800) 854-3805. In California (714) 774-1010.

Circle 80 on inquiry card.

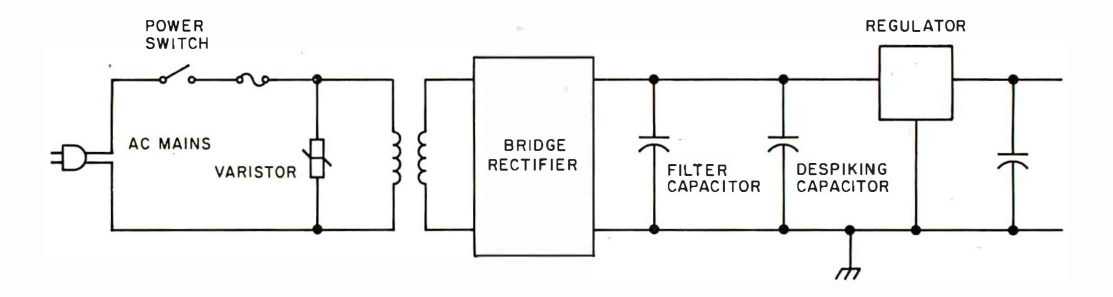


Figure 2: The combined isolation and shunting method is the best way to protect your system from voltage transients. The varistor shunts large transients in the AC source of power. Small high frequency "despiking" capacitors provide a low impedance path for any components of the external spike which make it through the transformer and rectifier. (The inductance of the regular filter capacitor tends to limit its usefulness at high frequencies.)

path to ground and pass by unnoticed. But more than likely it will enter some dandy appliance, or your computer, and do all the damage it can. Remember that 1 that mysteriously appeared in memory shortly after you wrote a 0? Have you ever wondered how that bad data got into your system? It could have been put there by your next door neighbor turning on a vacuum cleaner. You have seen rectifier diodes fail when they were carrying only a tenth of their rated current, voltage regulator integrated circuits die when they weren't even running warm, and transistors stop working when the hermetic seal broke, letting out the smoke. (I've always wondered how they work with all that smoke in there.) If you have mysterious errors in your system, transient and random, chances are a spike might have been involved.

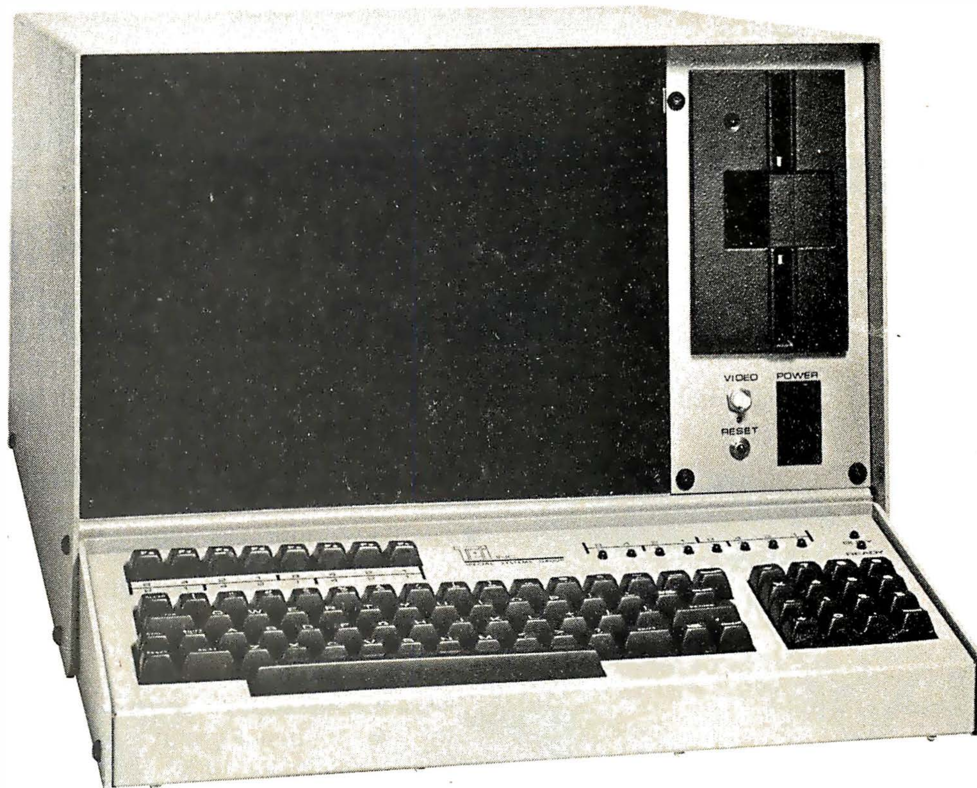
Now let's get to the good part: how to get rid of the little monsters. There are two basic techniques available. First, you can attempt to isolate the equipment from the source of the spikes by running it on batteries or an uninterruptible power supply. Isolation transformers show up at the surplus dealers occasionally, but are usually expensive. The second method is usually cheaper, but is somewhat less effective. Use the voltage divider principle and shunt the spike to ground through a low impedance at the power supply. A common example of this principle is the 0.01 μF capacitor placed between the power buses and ground of a digital circuit, to suppress the low level switching transients of digital integrated circuits. Since we are talking about transients that come in over AC lines, we need to put the low impedance on either the AC line or the power supply bus. On the DC side, hefty filter capacitors do this for the spikes with low frequency characteristics, but they often exhibit stray inductance which looks like a high impedance to a fast pulse. Putting a 0.01 μF capacitor in parallel with the filter capacitor will take care of many of these. Nonlinear devices such as spark gaps and varistors may be placed on the AC line. The last part of the shunt method is the most important. Put a good ground on the machine! If your house doesn't have three wire outlets, tie the case ground to a water pipe; if you have to, drive a ground rod. Be aware of the grounding system in all your electronic equipment. Poor grounding practice can cause shocks, ground loops, and erratic operation. *[When I took my system away from its usual solid grounding arrangements for a demonstration at the ACGNJ meeting May 20 of this year, the lack of a good ground became painfully obvious: programs which have never before committed suicide became quite distressed and recalcitrant . . .* CH*]*

We can expect to adequately protect the hardware without much trouble (or cash). The best procedure is to use a combination of the above methods as shown in figure 2.

I've tried to explain a little about voltage transients without getting into the physics of semiconductor failure or transient generation analysis. If you want to become better versed in this field, read several of the references. They all offer good background material and references 2 and 3 give detailed information. Hopefully, you are among the many who haven't had any problem with spikes. The best time to prepare for them is before they give you trouble.■

REFERENCES

1. Westinghouse Electric Corporation, *Electrical Transmission and Distribution Book,* East Pittsburgh PA.
2. G J Hahn and F D Martzloff, July 1970, "Surge Voltages in Residental Power Circuits," *IEEE Transactions on Power Apparatus and Systems,* 89 (6) 1049-1056.
3. General Electric Company, *Transient Voltage Suppression Manual,* Syracuse NY.



the Processor Terminal.

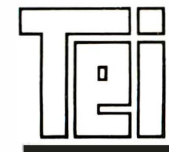
A logical forward step in Microcomputer design

the **Processor Terminal.** A new design by **TEI** and look at what you get . . . a complete, self contained microcomputer system with display and mass storage, a full keyboard and plenty of slot space for additional boards. And that famous **TEI CVT** power supply that makes brownouts a thing of the past.

Display — A 15" high-resolution black and white video display with an optical filter face plate to reduce glare and improve type visability . . . **Keyboard** — Full upper and lower case ASCII detached keyboard with 8 programmable special function keys. Keyboard status indicators show computer BUSY or READY. And a 16-key numeric cluster pad set up calculator style . . . **Disk Drive & Controller** — A Shugart SA-400 mini-floppy disk drive. Soft sectored with a capacity of about 90 KB. IBM compatible format. Controller will handle 3 drives . . . **CPU** — 8080 based with a flexible design that allows you to implement a start up "jump to" operation to any dip switch selected byte address you choose. Merely turn on power or press RESET and you are off and running. Excellent for power failure automatic restart . . . **Memory** — 16K of static RAM memory. Low power chips. Selectable address assignment and memory protect features . . . **I/O** — 3P+3S input/output board. 3 parallel ports and 3 serial ports with selectable baud rates of 75 to 19,200. RS-232C and TTL outputs . . . **Video** — A video board provides the support for the video display functions . . . **Mainframe** — A 12 slot mainframe with a 17-amp CVT power supply, motherboard assembly, heavy duty aluminum cabinet, fan and washable filter. All edge connectors and card guides provided . . . **Software** — CP/M disk operating system and BASIC provided on disk.

the **Processor Terminal** (Model MCS-PT) fully assembled and tested is priced at $3495.00.

the **Processor Terminal** partially assembled (We build the cabinet, keyboard, monitor, power supply, disk drive and motherboard and you build the CPU, RAM, I/O, Video and Controller boards which we supply as kits). This partially assembled unit is priced at $2995.00.



SPECIAL SYSTEMS GROUP

MCS
MICROCOMPUTER SYSTEM

Contact your local TEI Dealer or if you are not near one of our dealers, write or call CMC Marketing Corp direct for more information.

CMC MARKETING CORP
7231 Fondren Rd, Houston, TX 77036 Telephone (713) 774-9526

See you at Pers. Comp. 10/27-29 Chicago Booth # 53

Circle 23 on inquiry card.

Circle 141 on inquiry card.

Programming Quickies

Simple Math Lessons

Here is a program I wrote using Tom Pittman's Tiny BASIC. It originally appeared in *KIM-1 User's Notes*. This program allows my two children to play with the computer and also learn math. The output of the program looks like this:

```
THIS IS A MATH TEST
   12
X   6
   ---
    ?
```

If the correct answer is input, the computer replies with YOU'RE RIGHT and a new problem is set up. For a wrong answer the reply is ??WRONG??, TRY AGAIN and the same problem is repeated. If you answer incorrectly three times THE RIGHT ANSWER IS 72 appears and a new example is set up.

The actual problems are randomly chosen. The number limits for multiplication are set at line 200 for the multiplicand and 205 for the multiplier. Lines 305 and 355 define the two addends for addition.■

Robert G Lloyd
7554 Southgate Rd
Fayetteville NC 28304


```
10  PR "THIS IS A MATH TEST"
15  PR
20  LET V=0
30  LET I=0
35  LET Z=0
40  PR  "TYPE 1  FOR MULTIPLICATION"
50  PR
60  PR  "TYPE 2 FOR ADDITION"
70  PR
80  INPUT I
90  PR
100  IF I=1 GOTO 200
110  IF I=2 GOTO 350
120  IF D=Q GOTO 500
130  GOTO 600
190  END
200  LET X=(RND (12)+1)
205  LET Y=(RND (12)+1)
210  IF X <=10 GO TO 230
220  GOTO 240
230  PR "  ";X
235  GOTO 260
240  PR "  ";X
260  IF Y<=10 GOTO 280
270  GOTO 290
280  PR "  X  ";Y
285  GOTO 300
290  PR "X  ";Y
300  PR "____  ",
310  LET Q=X*Y
320  INPUT D
330  GOTO 120
350  LET X=(RND (50)+1)
355  LET Y=(RND (50)+1)
360  IF X<=10 GOTO 380
370  GOTO 390
380  PR "    "; X
385  GOTO 410
390  PR "   "; X
410 IF Y<=10 GO TO 430
420  GOTO 440
430  PR "  +  "; Y
435  GOTO 450
440  PR  " + "; Y
450  PR  "____",
460  LET Q=X+Y
470  INPUT D
480  GOTO 120
500  PR  "YOU'RE RIGHT"
505  PR
508  LET Z=Z+1
509  IF Z<3 GOTO 512
510  GOTO 10
512  IF I=1 GOTO 200
514  IF I=2  GOTO 350
600  PR  " WRONG  ,  TRY AGAIN"
610  PR
620  LET V=V+1
630  IF  V=3 GOTO 650
640  IF I=1 GOTO 210
645  IF I=2 GOTO 360
650  PR "THE RIGHT ANSWER IS  ",
655  PR Q
660  PR
670  GOTO 10
```



Look To The North Star HORIZON Computer.

HORIZON™— a complete, high-performance microprocessor system with integrated floppy disk memory. HORIZON is attractive, professionally engineered, and ideal for business, educational and personal applications.

To begin programming in extended BASIC, merely add a CRT or hard-copy terminal. HORIZON-1 includes a Z80A processor, 16K RAM, minifloppy™ disk and 12-slot S-100 motherboard with serial terminal interface — all standard equipment.

WHAT ABOUT PERFORMANCE?

The Z80A processor operates at 4MHZ — double the power of the 8080. And our 16K RAM board lets the Z80A execute *at full speed.* HORIZON can load or save a 10K byte disk program in less than 2 seconds. Each diskette can store 90K bytes.

AND SOFTWARE, TOO

HORIZON includes the North Star Disk Operating System and full extended BASIC on diskette ready at power-on. Our BASIC, now in widespread use, has everything desired in a BASIC, including sequential and random disk files, formatted output, a powerful line editor, strings, machine language CALL and more.

EXPAND YOUR HORIZON

Also available— Hardware floating point board (FPB); additional 16K memory boards with parity option. Add a second disk drive and you have HORIZON-2. Economical serial and parallel I/O ports may be installed on the motherboard. Many widely available S-100 bus peripheral boards can be added to HORIZON.

QUALITY AT THE RIGHT PRICE

HORIZON processor board, RAM, FPB and MICRO DISK SYSTEM can be bought separately for either Z80 or 8080 S-100 bus systems.

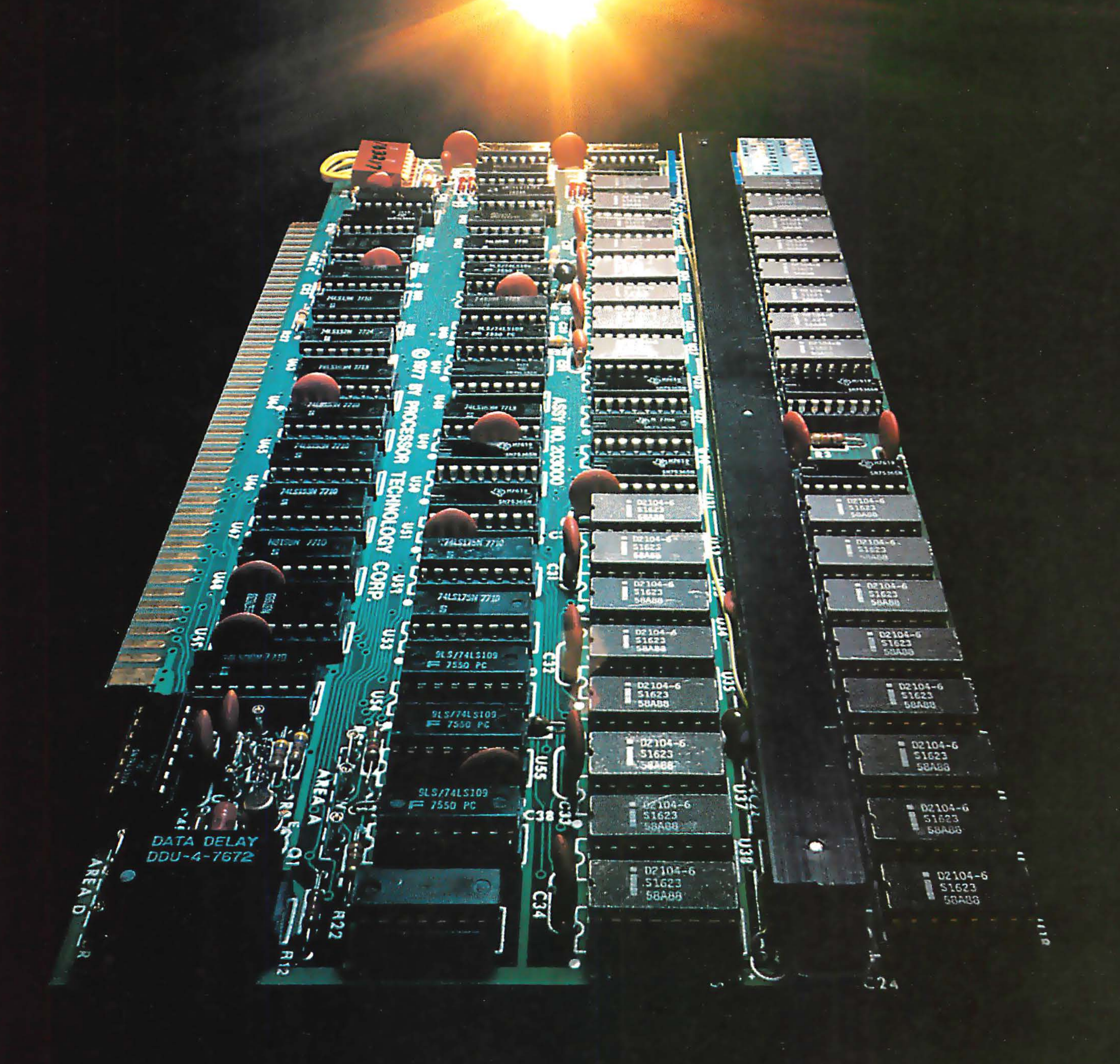
HORIZON-1 $1599 kit; $1899 assembled.
HORIZON-2 $1999 kit; $2349 assembled.

16K RAM—$399 kit; $459 assembled; Parity option $39 kit; $59 assembled. FPB $259 kit; $359 assembled. Z80 board $199 kit; $259 assembled. Prices subject to change. HORIZON offered in choice of wood or blue metal cover at no extra charge.

Write for free color catalogue or visit your local computer store.

NORTH STAR ★ COMPUTERS
2465 Fourth Street • Berkeley, California 94710 • (415) 549-0858

Circle 105 on inquiry card.


PROCESSOR TECHNOLOGY CORP
DATA DELAY
DDU-4-7672
AREA D
AREA A
9LS/74LS109
7550 PC
D2104-6
S1623
58A88
C38
C34
C24
R22
R12
U38
U55



The Small Computer

Twenty-five years ago a computer as powerful as the new Processor Technology SOL-20/8 priced out at a cool million.

Now for only $1350 in kit form or $1850 fully assembled and tested you can have your own small computer with perhaps even more power. It comes in a package about the size of a typewriter. And there's nothing like it on the market today. Not from IBM, Burroughs, DEC, HP or anybody else!

It fills a new role

If you're an engineer, scientist or businessman, the Sol-20 can help you solve many or all of your design problems, help you quantify research, and handle the books too. For not much more than the price of a good calculator, you can have high level computer power.

Use it in the office, lab, plant or home

Sol-20 is a smart terminal for distributed processing. Sol-20 is a stand alone computer for data collection, handling and analysis. Sol-20 is a text editor. In fact, Sol-20 is the key element of a full fledged computer system including hardware, software and peripheral gear. It's a computer system with a keyboard, extra memory, I/O interfaces, factory backup, service notes, users group.

It's a computer you can take home after hours to play or create sophisticated games, do your personal books and taxes, and a whole host of other tasks.

Those of you who are familiar with small computers will recognize what an advance the Sol-20 is.

Sol-20 offers all these features as standard:

8080 microprocessor — 1024 character video display circuitry — control PROM memory — 9216 words of static low-power RAM — 2048 words of preprogrammed PROM — built-in cassette interface capable of controlling two recorders at 1200 bits per second — both parallel and serial standardized interface connectors — a complete power supply including ultra quiet fan — a beautiful case with solid walnut sides — software which includes a preprogrammed PROM personality module and a data cassette with BASIC-5 language plus two sophisticated computer video games — the ability to work with all S-100 bus products.

Full expansion capability

Tailor the Sol-20 system to your applications with our complete line of peripheral products. These include the video monitor, audio cassette and digital tape systems, dual floppy disc system, expansion memories, and interfaces.

Write for our new 22 page catalog. Get all the details.

Processor Technology, Box B, 7100 Johnson Industrial Drive, Pleasanton, California 94566. Phone (415) 829-2600.

Processor Technology

See Sol here...

ALABAMA

ICP, Computerland
1550 Montgomery Hwy.
Birmingham, AL 35226
(205) 979-0707

ARIZONA

Byte Shop Tempe
813 N. Scottsdale Rd.
Tempe, AZ 85281
(602) 894-1129

Byte Shop Phoenix
12654 N. 28th Dr.
Phoenix, AZ 85029
(602) 942-7300

Byte Shop Tucson
2612 E. Broadway
Tucson, AZ 85716
(602) 327-4579

CALIFORNIA

The Byte Shop
1514 University Ave.
Berkeley, CA 94703
(415) 845-6366

Computer Center
1913 Harbor Blvd.
Costa Mesa, CA 92627
(714) 646-0221

DCI Computer Systems
4670 N. El Capitan
Fresno, CA 93711
(209) 266-9566

Bits 'N Bytes
679 S. State College Blvd.
Fullerton, CA 92631
(714) 879-8386

The Byte Shop
16508 Hawthorne Blvd.
Lawndale, CA 90260
(213) 371-2421

Opamp/Computer
1033 N. Sycamore Ave.
Los Angeles, CA 90038
(213) 934-3566

The Computer Mart
624 West Katella #10
Orange, CA 92667
(714) 633-1222

Byte Shop
496 South Lake Ave.
Pasadena, CA 91101
(213) 684-3311

Micro-Computer
Application Systems
2322 Capitol Avenue
Sacramento, CA 95816
(916) 443-4944

The Computer Store
of San Francisco
1093 Mission Street
San Francisco, CA 94103
(415) 431-0640

Byte Shop
321 Pacific Ave.
San Francisco, CA 94111
(415) 421-8686

The Byte Shop
2626 Union Avenue
San Jose, CA 95124
(408) 377-4685

The Computer Room
124H Blossom Hill Rd.
San Jose, CA 95123
(408) 226-8383

The Byte Shop
509 Francisco Blvd.
San Rafael, CA 94901
(415) 457-9311

The Byte Shop
3400 El Camino Real
Santa Clara, CA 95051
(408) 249-4221

Recreational Computer
Centers
1324 South Mary Ave.
Sunnyvale, CA 94087
(408) 735-7480

Computer Components
5848 Sepulveda Blvd.
Van Nuys, CA 91411
(213) 786-7411

The Byte Shop
2989 North Main St.
Walnut Creek, CA 94596
(415) 933-6252

Byte Shop
14300 Beach Blvd.
Westminster, CA 92683
(714) 894-9131

COLORADO

Byte Shop
2040 30th St.
Boulder, CO 80301
(303) 449-6233

Byte Shop
3464 S. Acoma St.
Englewood, CO 80110
(303) 761-6232

FLORIDA

Byte Shop of Miami
7825 Bird Road
Miami, FL 33155
(305) 264-2983

Microcomputer
Systems Inc.
144 So. Dale Mabry Hwy.
Tampa, FL 33609
(813) 879-4301

GEORGIA

Atlanta Computer Mart
5091-B Buford Hwy.
Atlanta, GA 30340
(404) 455-0647

ILLINOIS

Champaign Computer
Company
318 N. Neil Street
Champaign, IL 61820
(217) 359-5883

itty bitty machine co.
1316 Chicago Ave.
Evanston, IL 60201
(312) 328-6800

itty bitty machine co.
42 West Roosevelt
Lombard, IL 60148
(312) 620-5808

INDIANA

The Data Domain
406 So. College Ave.
Bloomington, IN 47401
(812) 334-3607

The Byte Shop
5947 East 82nd St.
Indianapolis, IN 46250
(317) 842-2983

Computers Unlimited
7724 East 89th Street
Indianapolis, IN 46256
(317) 849-6505

The Data Domain
7027 N. Michigan Rd.
Indianapolis, IN 46268
(317) 251-3139

IOWA

The Computer Store
of Davenport
616 West 35th Street
Davenport, IA 52806
(319) 386-3334

KENTUCKY

The Data Domain
3028 Hunsinger Lane
Louisville, KY 40220
(502) 456-5242

MICHIGAN

The Computer Store
of Ann Arbor
310 East Washington
Ann Arbor, MI 48104
(313) 995-7616

Computer Mart
of Royal Oak
1800 W. 14 Mile Rd.
Royal Oak, MI 48073
(313) 576-0900

General Computer Store
2011 Livernois
Troy, MI 48084
(313) 362-0022

MINNESOTA

Computer Depot, Inc.
3515 W. 70th St.
Minneapolis, MN 55435
(612) 927-5601

NEW JERSEY

Hoboken Computer Works
No. 20 Hudson Place
Hoboken, NJ 07030
(201) 420-1644

The Computer Mart
of New Jersey
501 Route 27
Iselin, NJ 08830
(201) 283-0600

NEW YORK

The Computer Mart
of Long Island
2072 Front Street
East Meadow, L.I. NY 11554
(516) 794-0510

The Computer Shoppe
444 Middle Country Rd.
Middle Island, NY 11953
(516) 732-4446

The Computer Mart
of New York
118 Madison Ave.
New York, NY 10001
(212) 686-7923

The Computer Corner
200 Hamilton Ave.
White Plains, NY 10601
(914) 949-3282

OHIO

Computer Mart of Dayton
2665 S. Dixie Ave.
Dayton, OH 45409
(513) 296-1248

OREGON

Byte Shop Computer Store
3482 SW Cedar Hills Blvd.
Beaverton, OR 97005
(503) 644-2686

The Real Oregon
Computer Co.
205 West 10th Ave.
Eugene, OR 97401
(503) 484-1040

Byte Shop Computer Store
2033 SW 4th Ave.
Portland, OR 97201
(503) 223-3496

PENNSYLVANIA

Byte Shop of
Delaware Valley
1045 Lancaster Pike
Bryn Mawr, PA 19010
(215) 525-7712

RHODE ISLAND

Computer Power, Inc.
M24 Airport Mall
1800 Post Rd.
Warwick, RI 02886
(401) 738-4477

TEXAS

Computer World
926 N. Collins
Arlington, TX 76011
(817) 469-1502

Byte Shop
3211 Fondren
Houston, TX 77063
(713) 977-0664

Computertex
2300 Richmond Ave.
Houston, TX 77006
(713) 526-3456

Interactive Computers
7646½ Dashwood Rd.
Houston, TX 77036
(713) 772-5257

Neighborhood Computer
Store
#20 Terrace Shopping Center
4902 - 34th Street
Lubbock, TX 79410
(806) 743-2787

The Micro Store
634 So. Central
Expressway
Richardson, TX 75080
(214) 231-1096

VIRGINIA

The Computer Systems
Store
1984 Chain Bridge Rd.
McLean, VA 22101
(703) 821-8333

Media Reactions Inc.
11303 South Shore Dr.
Reston, VA 22090
(703) 471-9330

The Home Computer Center
2927 Virginia Beach Blvd.
Virginia Beach, VA 23452
(804) 340-1977

WASHINGTON

Byte Shop Computer Store
14701 N.E. 20th Ave.
Bellevue, WA 98007
(206) 746-0651

The Retail Computer Store
410 N.E. 72nd
Seattle, WA 98115
(206) 524-4101

WISCONSIN

Madison Computer Store
1910 Monroe St.
Madison, WI 53711
(608) 255-5552

The Milwaukee
Computer Store
6916 W. North Ave.
Milwaukee, WI 53213
(414) 259-9140

CANADA

Trintronics
160 Elgin St.
Place Bell Canada
Ottawa, Ontario K2P 2C4
(613) 236-7767

First Canadian
Computer Store Ltd.
44 Eglinton Ave. West
Toronto, Ontario M4R 1A1
(416) 482-8080

The Computer Place
186 Queen St. West
Toronto, Ontario M5V 1Z1
(416) 598-0262

Pacific Computer Store
4509-11 Rupert St.
Vancouver, B.C. V5R 2J4
(604) 438-3282

Processor Technology

Circle 117 on inquiry card.

My Experiences with the 2650

A Report from Our 14 Year Old Correspondent

Brian K Moran
7335 N Manning Dr
Peoria IL 61614

About the Author

Brian K Moran has the honor of being the youngest BYTE author to date. Brian is a 14 year old student at Richwoods High School in Peoria IL. His achievements include winning first place in science fairs on the school, regional and state levels with a project concerning computers. Brian presently has a working AMT-2650, and is designing his own computer based on the 2650 processor.

When I saw an ad in *Electronics* magazine for the Signetics 2650, I had a "sixth sense" that this was the processor I wanted. After contacting Signetics Corporation I received the 2650 manual. I had only started to learn about computers two months before, so I did not understand everything in the manual. I had no one to ask; my mom and dad are not familiar with computer technology. I began to write to Signetics, asking about various things, and they wrote back expressing much enthusiasm about my being interested in computers at such a young age. (I was 13 years old).

Signetics made available to me a 3 day seminar about the 2650 and about microcomputers in general. Needless to say, I was ecstatic. Even my parents were excited! When I arrived at the sales office where the seminar was to be held, I found I was the only person under 20 years of age. There was one person from a well-known megacomputer company, three men from a well-known amusement device company, two instructors, and myself. These adults were surprised that a "kid" would be learning about computers, and they asked me many questions.

The first day of the seminar went well, considering that my specialty is hardware, and I actually began to understand software. I had many chances to discuss certain aspects of personal computers with the man from the megacomputer company, and over lunch we discussed many problems of systems going berserk, dropping bits, breaking down, etc.

The second day we studied the problem of programming IO ports, and tested our programs on a timeshare computer service provided by the seminar. My program (the first I had written) had one bug in it. The problem was to read in data from a certain port into a specified register and output it to

a certain port containing an LED for each data line. The program was to do this continually, but when I loaded the data in from the port I forgot to clear it first.

I wanted to keep the program to an absolute minimum because the Teletype I was using kept losing contact with the timeshare computer. After fighting a battle of trying to write and save programs before the modem "crashed," the final score was: modem 4, me 1. I finally finished it. At this point the instructor said I could use another Teletype. No way!

Now came my turn to load my program into the demonstration computer. The 2650 must have liked me because it worked right after I loaded the program and pressed reset.

The third and last day at the seminar we learned about hardware usage and interfaces. I was sad to be leaving when it came time to bid everyone goodbye. I had become good friends with all the people and they had all helped me in one way or another.

The seminar was in March, and until late May I programmed on paper since I had no computer, nor access to one. I decided to purchase an AMT-2650 from Applied Microtechnology so I could learn more about it before I designed and built my own processor board. It was two months and five days from the date of my order that my computer was delivered. It arrived the day before school reopened. This was a great disappointment because I was planning to work on it during summer vacation.

After programming the diagnostics to check out the computer, I discovered that bit 0 in output port C remained lit when the computer was in the run state, and when a true bit was in position 0 in output port C, the bit in the data load byte would come on, making things more confusing.

Despite all the bugs, I developed many short programs on this computer including one that rotates left one bit in output port C until it gets to bit 7, while another bit in output port D rotates right at the same speed; then both would repeat. One row of LEDs is on top of anther, so that, when this program is run, the lights seem to chase each other in circles. There is one catch: the lights go very fast at first and get slower and slower until they come to a full stop and the machine halts. Upon reset, the whole process is started again.

I'm still listing features I want for my processor board and front panel. If anyone is interested in the 2650 please contact me, since no one I know uses this processor, and I would like to possibly start a users' group.■

Does Anybody Know What Time It Is?

Robert Grappel
148 Wood St
Lexington MA 02173

One of the earliest products of LSI technology that filtered down to the hobbyist was the "clock chip." This little "beauty" divided the 60 Hz line signal down to seconds, minutes and hours. . .and displayed the results on 7 segment LED or other displays. Today these "clocks" come in a great variety of types, sizes and functions. They come tiny for watches. Some have extra timers and alarm capabilities. They are inexpensive, and require little in the way of external circuitry. For long term timing applications, they form an ideal solution for computer experimenters.

For many personal computer applications, it would be useful for the computer to have a knowledge of the time. The computer can certainly count interrupts from a crystal time standard, but why not use external hardware optimized for the timekeeping function, ie: a "clock chip?" This article describes an approach to such a linking of computer and clock. The clock I used had a National Semiconductor MM5314, but other clock chips using multiplexed 7 segment displays will also work. The circuit attaches to the display lines, does not disable the clock functions or the display, and is easily added inside the clock's case. It simply lets the computer read the clock digits (with the appropriate software) at the same time that the ordinary electronic display is produced.

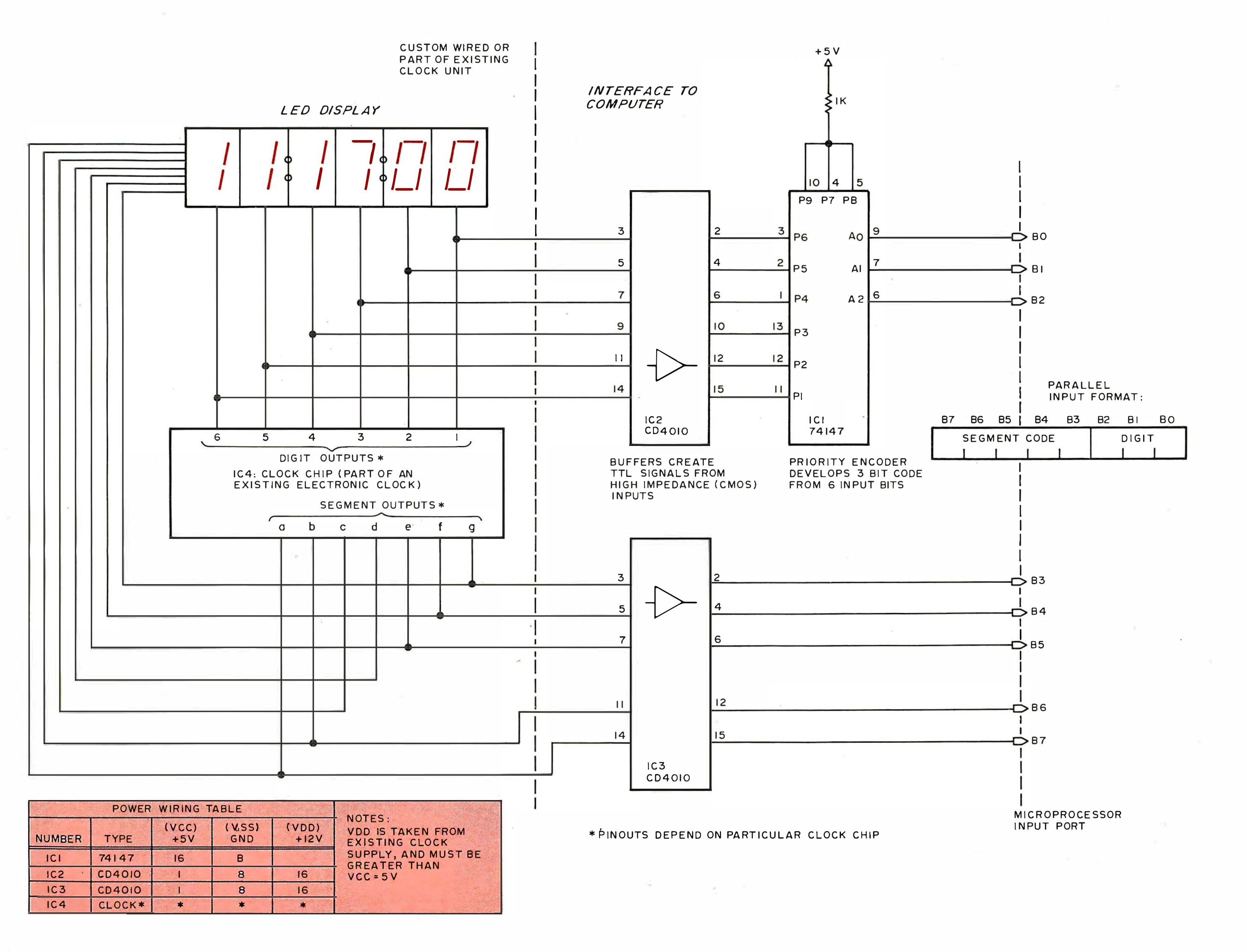
The hardware interface is shown in figure 1. It consists of three integrated circuits at a total cost of less than $5. Two CD4010 buffers are used to convert the MOS voltage levels of the clock to TTL levels. These buffers are CMOS, so they form almost no load on the clock circuits. The pins labelled V_{DD} are tied to the clock supply. The pins labelled V_{CC} are tied to the computer TTL power supply of 5 V. The common ground line for clock and interface and computer is V_{SS}. The only criteria assumed here are that V_{SS}=GND< V_{CC}=+5 V < V_{DD}.

The clock uses a multiplexed 7 segment display format. This means that each digit is formed from seven data bits, and the digits are sequenced one at a time. The lower buffer works on the segment signals. Although seven bits are used for the display, only five are needed to uniquely decode digits. This circuit sends the a, b, e, f and g signals to the computer. These five bits are used in a software table lookup to convert to the digit code. The buffer IC2 handles the digit signals. The six digits are scanned right to left, from seconds digit to tens of hours digit. These six signals are converted to a 3 bit binary number by a 74147 priority encoder. Since both the clock and the 74147 utilize inverted logic, the connections have been manipulated to provide a normal logic output. (Seconds digit is 1, tens of seconds is 2, etc). Thus each digit is converted to eight data bits: three which describe its place in the display and five when uniquely map to its value.

The subroutine of listing 1 illustrates how to read the clock interface. It is written for a Motorola 6800, but should be readily convertible to other processors. The location CLKIO is the interface input (which is assumed to be previously initialized if it is a PIA data location). The subroutine reads the digits from right to left and stores the ASCII code for each digit in a 6 byte storage area. This area is pointed to by the X register contents when the subroutine is called.

The code between WAITD and CLK2 continuously samples the interface waiting for the low order 3 bit digit code pointed to by the B register. When data for that digit is presented, its segment data is separated from the input value and those five bits (shifted right three positions) are used in a table lookup in SEGTAB. This returns the ASCII digit. If no digit corresponds to the bit pattern (hardware error), the letter E is returned. This ASCII character is stored in the storage area. The routine loops through all 6 digit locations and then returns.

Figure 1: Schematic of the clock interface, and a partial schematic of the clock chip and display circuitry. The interface circuitry is intended to convert the signals from an existing electronic clock using MOS integrated circuits and LED displays (left) into TTL compatible levels usable by the microprocessor port at right. Some analysis of the particular clock used is required to attach the interface wires to the appropriate digit and segment output lines. Since the CMOS DC4010 level shifting buffers employed have high impedance inputs, the loading of the clock chip's output lines will not affect operation of the clock itself when the computer is attached.



POWER WIRING TABLE

NUMBER	TYPE	(VCC) +5V	(VSS) GND	(VDD) +12V
IC1	74147	16	8	
IC2	CD4010	1	8	16
IC3	CD4010	1	8	16
IC4	CLOCK*	*	*	*

```
* PROGRAM TO READ CLOCK PERIPHERAL HARDWARE
* CALL SUBROUTINE WITH ADDRESS OF 6-BYTE STORAGE
* AREA POINTED TO IN X-REGISTER
* ASCII DIGITS WILL BE STORED THERE
* ORDER IS:
* SECONDS, 10'S OF SECONDS
* MINUTES, 10'S OF MINUTES
* HOURS, 10'S OF HOURS
* HARDWARE ERRORS WILL STORE CHAR. 'E'
*
* WRITTEN BY R. D. GRAPPEL
* JANUARY 1977
* FOR MOTOROLA 6800 PROCESSOR
*
CLOCK    LDA B #1  START WITH SECONDS DIGIT
WAITD    LDA A CLKIO     READ CLOCK PORT
         PSH A   SAVE DATA
         AND A #7  GET SEGMENT VALUES
         CBA
         BEQ CLK2   CORRECT DIGIT
         PUL A
         BRA WAITD WAIT FOR DIGIT
CLK2     PUL A  GET SEGMENT VALUES
         LSR A
         LSR A
         LSR A
         STX SAVE SAVE X-REGISTER
         LDX #SEGTAB    POINT TO CONVERSION TABLE
         STA A INDEX+1  MODIFY NEXT INSTR.
INDEX    LDA A 0,X GET ASCII CODE
         LDX SAVE  RESTORE X-REGISTER
         STA A  0,X STORE CHARACTER
         INX
         INC B      MOVE POINTERS
         CMP B #7   DONE WITH 6 DIGITS?
         BNE WAITD  LOOP UNTIL DONE
         RTS
*
SAVE     RMB 2      TEMPORARY SAVE AREA
*
SEGTAB   FCC 'EEEE'      7-SEGMENT TO ASCII CONVERSION TABLE
         FCC 'EE16'      'E' MEANS NO DIGIT HAS THIS PATTERN
         FCC '1EE4'      5-BIT INDEX INTO TABLE FORMED FROM
         FCC 'EEEE'      SEGMENTS A,B,E,F,G
         FCC 'EEE5'     (A SEG. IS HIGH ORDER)
         FCC 'EEEE'
         FCC '73E9'
         FCC 'E208'      HANDLES ALL FORMS OF DIGITS
```

Listing 1: A program written for the 6800 which will translate the outputs of the clock chip at the input port CLKIO into a 6 byte string of ASCII digits. Due to the typical scanning times of clock displays, the execution of this routine will complete in 6 to 11 milliseconds, so use in time-dependent portions of a program may require careful thinking.

The table lookup is done with the trick of instruction modification at INDEX. If this offends your sense of "proper programming practice," then try the code used in the MORSER article (BYTE, October 1976, page 34).

The clock steps through digits at a roughly 1 kHz rate. Since the clock and the computer are not synchronized, it might take up to 11 digit times for the program to run to completion. The subroutine thus executes in between 6 and 11 ms. It requires about 80 bytes of memory.

Now there is no excuse for your computer not to know the time unless its clock stops.■

What you should know about our "workhorse micro" The Peripheral Universal Processor



We have done it again - yes, Seal's has designed and engineered the only truly continuous-duty micro computer.

We are proud to announce the arrival of the newest member to the Seals' family of performance products... **the PUP-1 Peripheral Universal Processor.** The PUP-1 was designed with the same reliability and attention to detail that has enabled Seals' products to set the quality standards for the industry. The PUP-1 was designed from the ground up to meet all the needs and demands placed on a continuous-duty micro computer by OEM applications.

The standard software package for the PUP-1 includes **DOS, EXTENDED BASIC, SAMPLE BUSINESS,** and **DISC OPERATING ROUTINES.** To make repairs more simple, should a malfunction occur, the PUP-1 comes with a **diagnostic program** that will locate the exact area of dysfunction.

The PUP-1 is fully operational in temperatures from **0-55° C** and can run on AC line voltages available anywhere in the world.

Designed to deliver reliability, performance and maintenance-free operation, the PUP-1 makes optimum cost effectiveness available for you.

PUP-1 SPECIFICATIONS

PROCESSOR
8-bit Z-80 CPU 2.5 MHz standard 4MHz option/other CPUs available.

STANDARD SOFTWARE
EXTENDED BASIC, DOS, diagnostics, sample business and disc operating routine.

BUS
Completely S-100 Compatible. All accepted signals including 'P-sync' and 'Refresh'. Plug-in terminators on 16 address lines, 8 data in, and 8 data out lines.

MEMORY
32K standard. Expandable to 0.5 Megabyte without additional power or extended mother-board.

DISK DRIVE
Built-in dual 'Shugart' mini-floppy standard. 86K per disk. Also available without disk or with one drive only (Double Density Coming).

I/O
Standard ports: 2 serial, 2 parallel. Each completely independent (any port may be run at any address). Addresses switch-selectable with software override. 50 to 19,200 baud.

POWER SUPPLY
Full-load rating + 8V @ 23A, + 18V @ 6A, -18V @ 6A
Input 95 - 125V ac and 195 - 250V ac, 50-60Hz,
Oversize rectifiers 50A + 8V, 25A ± 16V

FRONT PANEL
Push Buttons: 'INITIALIZE', 'RESET', 'STOP'
Indicators: 'RUN', 'WAIT', 'SINP', 'SOUT'.
Security Lock: 'OFF', 'INITIALIZE', 'RUN', 'PROTECT'.

REAR PANEL
9 cutouts for standard 25 pin EIA connector. 2 cutouts for standard 37 pin EIA connector. 2 cutouts for standard 15 pin EIA connector. 2 D holes for BNC connectors. Off-On Switch (main power). Fused main power.

MECHANICAL
Free-standing (rack mounting optional)
11-connector mother-board. Rigid main chassis.
Removable front panel.
Card-cage containing all components, except power supply and listed front and rear panel controls and connectors, removable for servicing.

OPERATING TEMPERATURE
Full specification at ambient 0-55°C.

ATTENTION SYSTEMS DESIGNERS:
Seals Electronics Inc., has developed a hybrid BASIC which combines the best features of both compiler and interpreter languages. Call factory direct or write Seals today for information explaining how our fundamental software program will assist you in your systems construction.

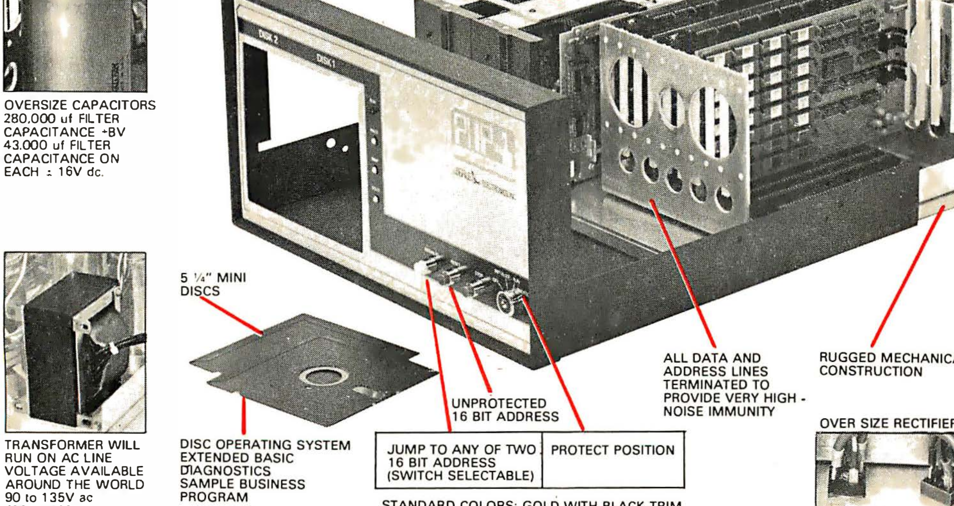


10728 DUTCHTOWN RD., CONCORD, TN. 37922
(615) 966-8771 TELEX NO. 55-7444



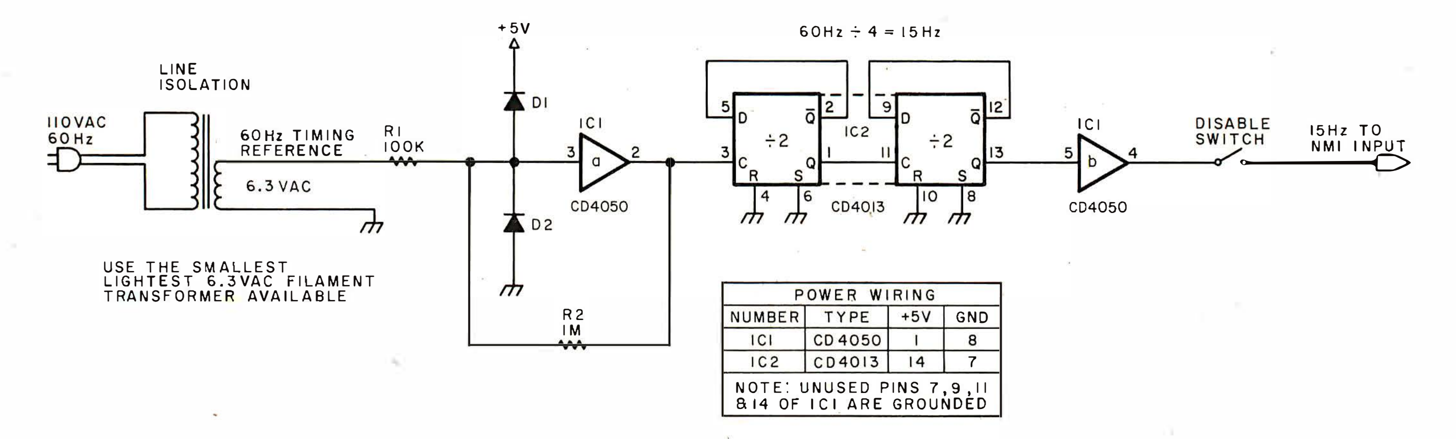
OVERSIZE CAPACITORS
280,000 uf FILTER CAPACITANCE +8V
43,000 uf FILTER CAPACITANCE ON EACH ± 16V dc.

TRANSFORMER WILL RUN ON AC LINE VOLTAGE AVAILABLE AROUND THE WORLD
90 to 135V ac
190 to 260V ac
50/60 Hz INPUT



STANDARD COLORS: GOLD WITH BLACK TRIM
OTHER COLORS AVAILABLE

Figure 1: A simple circuit which processes a 6.3 VAC reference signal derived from the power companies' 60 Hz grid to produce a digital logic level square wave at 15 Hz which can drive an interrupt line of a typical processor. The disable switch is optional and can be left out if the interrupt handlers are permanently loaded in ROM; otherwise, interrupts must be manually disabled while the systems software is bootstrapped into volatile memory.



POWER WIRING

NUMBER	TYPE	+5V	GND
IC1	CD 4050	1	8
IC2	CD 4013	14	7

NOTE: UNUSED PINS 7, 9, 11 & 14 OF IC1 ARE GROUNDED

Adding an Interrupt Driven Real Time Clock

James R Sneed
13831 NE 8th, Apt 86
Bellevue WA 98005

Whenever a computer is interacting with the real world, either through sensors or actuators, a real time clock can be valuable. Using a real time clock, the computer can run programs at specified times or intervals, or the computer may record the times at which events are sensed.

There are two basic types of real time clocks used in computing systems: the external (hardware) clock and the internal (software) clock. An external clock uses hardware to keep track of time, and periodically or on command transmits the time to the computer. *[Robert Grappel's article on page 68 of this issue shows one approach to such a clock . . .CH]* An internal software clock has hardware which interrupts the computer at regular intervals, and software which keeps track of time by incrementing a register whenever the computer receives a timing interrupt.

The hardware clock imposes a small software burden on the computer, and being separate from the computer, it need not be reset whenever the computer is shut off. The software clock imposes a larger software burden on the computer, and the clock must be initialized if the computer has been completely halted or had its power shut off. In applications where the computer operates continuously, the advantages of the software clock due to hardware simplicity outweigh its disadvantages due to increased software burden, and the software clock is the logical choice for a real time clock.

There are two key considerations involved in selecting the interrupt rate for the software clock. First, where the interrupt clock is derived by dividing a higher frequency clock, such as a 1 MHz computer clock, hardware simplicity favors as high an interrupt rate as possible, but the computational overhead of interrupt response increases with increasing interrupt rate. Second, a low interrupt rate produces a low computational burden but decreases timekeeping resolution and programming flexibility. Since my system requires no routines to be performed more often than 15 times per second, I decided that a 15 Hz interrupt derived by dividing the 60 Hz power line frequency by 4 would be an adequate interrupt rate. This gives a minimum event to event resolution of 67 ms.

Listing 1: Interrupt handler. This routine contains the overhead needed to field an NMI interrupt on a 6502 processor, save the state of the processor, call an interrupt processing subroutine, restore the state of the processor, and return from the interrupt event. If the jump at location 206 is replaced by NOP operations, this program will spin its wheels 15 times a second, doing nothing in response to the 15 Hz signal produced by the circuit of figure 1. With the exception of the JSR at location 206, this routine is independent of the location in memory of the software discussed in this article.

Hexadecimal Address	Hexadecimal Code	Op	Commentary
0200	48	PHA	Push accumulator onto stack
0201	8A	TXA	Transfer X register to accumulator
0202	48	PHA	Push X register onto stack
0203	98	TYA	Transfer Y register to accumulator
0204	48	PHA	Push Y register onto stack
0206	20 00 00	JSR	Call CLOCK
0209	68	PLA	Pull Y register from stack
020A	A8	TAY	Transfer accumulator to Y register
020B	68	PLA	Pull X register from stack
020C	AA	TAX	Transfer accumulator to X register
020D	68	PLA	Pull accumulator from stack
020E	40	RTI	Return from interrupt
FFFA	00 02		Interrupt address vector

The circuit in figure 1 produces the 15 Hz interrupts. The 60 Hz signal is taken from the secondary of a 6.3 V filament type transformer. (The term is a hangover from vacuum tube days when many tubes had 6.3 V filaments somewhat like incandescent light bulbs). The input to IC1A, a CMOS buffer, is clamped between 5 V and ground by diodes D1 and D2, which can be any silicon small signal diodes at hand. Resistor R2 provides positive feedback to produce about a half a volt of hysteresis in the switching of the buffer. This hysteresis reduces false interrupts due to line voltage fluctuations and transients. The two D type flip flops in IC2 are used as cascaded divide-by-two circuits. The 15 Hz output from IC2 is buffered to drive TTL loads by IC1B. To prevent runaway power consumption and the resulting chip destruction, the unused inputs of the CMOS integrated circuits are grounded.

The nonmaskable interrupt of the 6502 is edge triggered; that is, the processor receives an interrupt whenever the voltage on the nonmaskable interrupt line goes from high (>2.4 V) to low (<2.4 V). The nonmaskable interrupt line can then stay low without generating another interrupt. When the processor receives a nonmaskable interrupt it jumps to the memory address stored at FFFA and FFFB, and pushes the address from which it was interrupted and the processor status onto the stack so that it can return to the preinterrupt computation as soon as it has processed the interrupt. A switch is shown between the 15 Hz interrupt and the NMI line so that interrupts can be disabled after power is applied until the interrerupt handler for NMI has been loaded in volatile memory. If the interrupt handler is in read only memory, this switch can be omitted.

The contents of the accumulator and the X and Y registers should be saved by software when the interrupt is received and control switches to the interrupt handler program. This is done by pushing them onto the stack using appropriate instructions. Once the preinterrupt state has been safely preserved, the processing done as a result of the interrupt is performed. After the interrupt program has been completed, the preinterrupt contents of the Y and X registers and the accumulator are restored by pulling them off the stack. The processor then pulls the preinterrupt processor status and program address from the stack and returns to the previous computation. Listing 1 is a sample interrupt handler.

Listing 2 is a 24 hour clock generated in software by accumulating 15 Hz interrupts. This program contains only relative jumps and so is easily relocatable, either in volatile memory, EROM or PROM.

The operation of the program real time

Listing 2: Time of day clock. If the jump at line 206 in the interrupt handler of listing 1 references the CLOCK routine, locations C4 to C7 in memory address space are continously updated with hours, minutes, seconds and 1/15 seconds respectively as the 15 Hz interrupts invoke its action. The 6502 code of this routine has been constructed to use relative branches only, so that it can be relocated anywhere in memory address space at the convenience of its user without modification of the object code.

CLOCK (Real Time Clock)

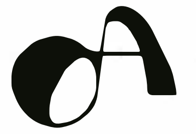
Hexadecimal Address	Hexadecimal Code	Label	Op	Operand	Commentary
0000	F8	CLOCK	SED		Set decimal mode
0001	18		CLC		Clear carry
0002	A5 C7		LDA	FSEC	Load seconds fraction
0004	69 01		ADC	1	Incr seconds fraction
0006	85 C7		STA	FSEC	Store seconds fraction
0008	38		SEC		Set carry
0009	E9 15		SBC	15	Subtract 15
000B	D0 2C		BNE	END	If not 15, go to end
000D	85 C7		STA	FSEC	Reset seconds fraction
000F	A5 C6		LDA	SEC	Load seconds
0011	18		CLC		Clear carry
0012	69 01		ADC	1	Incr seconds
0014	85 C6		STA	SEC	Store seconds
0016	38		SEC		Set carry
0017	E9 60		SBC	60	Subtract 60
0019	D0 1E		BNE	END	If not 60, go to end
001B	85 C6		STA	SEC	Reset seconds
001D	A5 C5		LDA	MIN	Load minutes
001F	18		CLC		Clear carry
0020	69 01		ADC	1	Incr minutes
0022	85 C5		STA	MIN	Store minutes
0024	38		SEC		Set carry
0025	E9 60		SBC	60	Subtract 60
0027	D0 10		BNE	END	If not 60, go to end
0029	85 C5		STA	MIN	Reset minutes
002B	A5 C4		LDA	HOURS	Load hours
002D	18		CLC		Clear carry
002E	69 01		ADC	1	Increment hours
0030	85 C4		STA	HOURS	Store hours
0032	38		SEC		Set carry
0033	E9 24		SBC	24	Subtract 24
0035	D0 02		BNE	END	If not 24, go to end
0037	85 C4		STA	HOURS	Reset hours
0039	D8	END	CLD		Clear decimal mode
003A	60		RTS		Return
00C4		HOURS			Storage for hours
00C5		MIN			Storage for minutes
00C6		SEC			Storage for seconds
00C7		FSEC			Storage for seconds/15

CLOCK is straightforward. Time is stored in BCD in the first page of memory: hours in 00C4, minutes in 00C5, seconds in 00C6, and 1/15 seconds in 00C7. When an interrupt is received and the preinterrupt state saved, the interrupt handler will call the real time CLOCK at 0000 (location 0206 in listing 1). The second's fraction is incremented and compared to 15. If it is less than 15 the processor will jump to the end of the clock program for return, but if it equals 15 the second's fraction is reset to zero and the seconds are incremented. Seconds, minutes and hours are handled similarly, counting modulo 60, 60 and 24 respectively. At the end of the program the processor returns to the interrupt handler. The clock can be set simply by loading the desired time into the time memory locations.

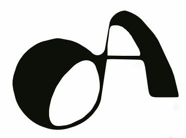
By comparing desired program times with the time of the real time CLOCK program, the processor may perform programs at any desired interval, up to one day, which is expressable as a multiple of 1/15 second. As an example, a program to be performed once per second would be executed only at those times when CLOCK has counted the second's fraction equal to zero.

It is important that the real time CLOCK should not impose an unreasonable computational burden on the computer. Using a 15 Hz interrupt and the program shown here, this criterion is satisfied. When run in a computer using a 6502 processor with a 1 MHz clock, the interrupt service requires about 1100 μs per second. This 0.1% cannot be called an excessive burden on the computer.■

OSBORNE & ASSOCIATES, INC.

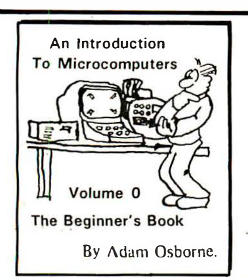


The World Leaders In Microprocessor Books



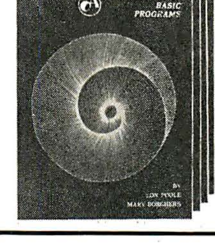
Many books on microprocessors and their use are now on the market, and most of them have names that sound alike. But Osborne & Associates' books have dominated this market since 1975, when our first book appeared. With rave reviews from all over the world — with more than five hundred university text adoptions, our books are all best sellers. In fact, "An Introduction To Microcomputers: Volume I — Basic Concepts" now holds the world's record in sales volume for any textbook sold for a profit.

If you want information on microprocessors, begin with the Osborne books.



"An Introduction To Microcomputers: Volume 0 — The Beginner's Book" By Adam Osborne.

This is the book for the absolute beginner. Assuming that you know nothing about computers, math or science of any kind, this book explains what computers are all about — and it takes you to the point where you can read Volume I. 300 pages.
Book No: 6001 $7.50



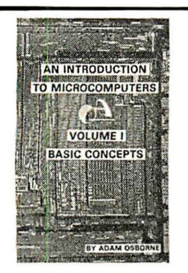
For the microcomputer user, a series of books provide complete programs, written in BASIC. All these books are by Lon Poole and Mary Borchers.

"Some Common Basic Programs" 200 pages.
Book No: 21002 $7.50

"Payroll With Cost Accounting — In Basic". 400 pages Book No.: 22002 $12.50

"Accounts Payable and Accounts Receivable"
Book No.: 23002 $12.50 Available November 30, 1977

"General Ledger System"
Book No.: 24002 $12.50 Available December 31, 1977



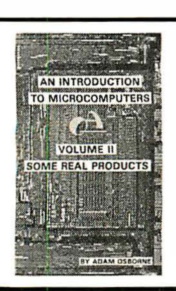
"An Introduction To Microcomputers: Volume I — Basic Concepts" By Adam Osborne.

The world's best selling textbook. This book explains, clearly, concepts common to all microcomputers, yet specific to none. 350 pages.
Book No.: 2001 $7.50

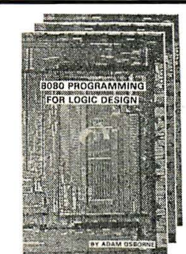
8080A
AND 8085
ASSEMBLY
LANGUAGE
PROGRAMMING
BY LANCE LEVENTHAL

"8080A and 8085 Assembly Language Programming" By Lance Leventhal.

This book is for the assembly language programmer or student; it explains assembly language programming for the 8080A and 8085 microcomputers. The book contains numerous examples. 400 pages.
Book No.: 31003 $7.50 Available October 31, 1977



"An Introduction To Microcomputers: Volume II — Some Real Products" (Revised June 1977) By Adam Osborne, Susanna Jacobson and Jerry Kane.
This book describes every common microprocessor and all of their support devices. Information is new and clearly written. Only data sheets are copied from manufacturers. 1200 pages.
Book No.: 3001 $15.00



The "Programming For Logic Design" series of books show how to use microprocessors in a digital logic environment.

"8080 Programming For Logic Design" By Adam Osborne. 300 pages.
Book No.: 4001 $7.50

"6800 Programming For Logic Design" By Adam Osborne. 300 pages.
Book No.: 5001 $7.50

"Z80 Programming For Logic Design" By Adam Osborne and Susanna Jacobson.
Book No: 7001 $7.50 (Available November 30, 1977)

OSBORNE & ASSOCIATES, INC. P.O. BOX 2036 BERKELEY, CA 94702 DEPT. I

TITLE	UNIT PRICE	QUANTITY
Volume 0 — The Beginner's Book (#6001)	$7.50 ea.	
Volume I — Basic Concepts (#2001)	$7.50 ea.	
Volume II — Some Real Products (#3001) revised 1977	$15.00 ea.	
8080 Programming For Logic Design (#4001)	$7.50 ea.	
6800 Programming For Logic Design (#5001)	$7.50 ea.	
8080A/8085 Assembly Language Programming (#31003)	$7.50 ea.	
Some Common BASIC Programs (#21002)	$7.50 ea.	
Payroll With Cost Accounting (#22002)	$12.50 ea.	

We will only invoice for purchase orders of over 10 books.
Shipping charges for bulk orders to be arranged.

○ Check or Money Order enclosed (Calif. residents include sales tax)

I will be using these books for ____________

NAME/COMPANY ____________

ADDRESS ____________

CITY, STATE AND ZIP ____________

TELEPHONE ____________

Please check one space below:

- ○ 4th Class Mail Delivery (3-4 weeks within the U.S.A.)
- ○ I have included $.50 per book for U.P.S. (allow 10 days)
- ○ I have included $1.50 per book for special rush shipment by air.
- ○ I have included $.50 per book foreign surface mail.
- ○ I have included $3.00 per book for foreign airmail.
- ○ I require information on consignments, discounts and distributors outside the U.S.A.

Circle 109 on inquiry card.

Floating Point Arithmetic

Burt Hashizume
POB 447
Maynard MA 01754

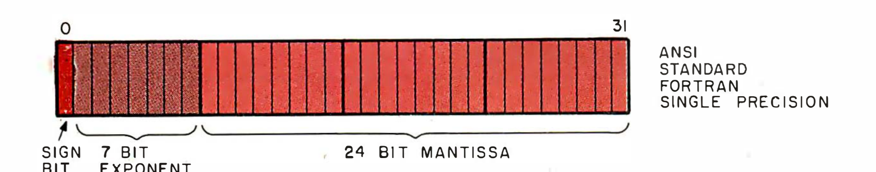


Figure 1: The American National Standards Institute (ANSI) floating point format for FORTRAN. It consists of a 24 bit mantissa, a 7 bit exponent and a sign bit.

Many computer hobbyists are finding 8 bit integer arithmetic inadequate for a variety of mathematical applications. 16 and even 32 bit fixed point calculations are being used with increasing frequency because of their greater accuracy. However, these techniques are still inherently inadequate for calculations performed over a wide range of numbers.

Using a 16 bit integer format, only numbers from 0 to 65,535 can be represented. Larger or smaller numbers can be represented by moving the implicit radix point, but the range of discrete values still remains constant. The fractional part of the quotient in a division of one large number by another could be lost.

If one could dynamically slide the radix point, the number range would be dramatically increased. Using the same format, very small fractions and very large integers can be represented as floating point numbers. This is made possible by keeping track of the radix point's position separately with an exponent.

Floating Point Formats

There are many ways to represent floating point numbers, but there are only three basic formats; the others are variations. Two of these (the dominant ones in the traditional computer industry) use different binary representations. The third format, the one with the most variations, uses a binary coded decimal (BCD) representation, and is widely used in the electronic calculator and home computer industry.

The first format, shown in figure 1, is used in American National Standards Institute (ANSI) FORTRAN. It consists of a 24 bit mantissa, a 7 bit exponent and a sign bit.

The mantissa represents a fraction with the radix point assumed to be to the left of the most significant digit. The exponent is in excess-64 notation, which is a 7 bit two's complement notation with the sign bit inverted, eg: a zero exponent (16^0) is 100 0000, the minimum exponent (16^{-64}) is 000 0000, and the maximum exponent (16^{-63}) is 111 1111. The algebraic sign bit of the value is associated with the mantissa, and the exponent's sign is inherent in its format: a one sign bit indicates the number is negative, and a zero sign bit indicates a positive number.

This is the data storage format of floating point numbers. All such data is assumed to be normalized (ie: the most significant digit in the mantissa is nonzero unless the number itself is zero, in which case all 32 bits are zero). Before a calculation, the numbers are assumed normalized; after a calculation they are normalized in the floating point accumulator before being stored.

The actual calculations take place in the floating point accumulator and other floating point registers. These registers can be in the hardware or in memory (software). Hardware floating point registers (expensive, but much faster than software) are used by large computers and many minicomputers, whereas most small computers implement floating point in software to keep costs down.

With the ANSI format a "guard byte" is used in the floating point registers to maintain accuracy in performing the calculations. The guard byte (see figure 2) is an 8 bit extension to the least significant end of the 24 bit mantissa, temporarily creating a 32 bit mantissa during calculations. By keeping track of 32 bits of accuracy throughout the operation, significance will not be lost when storing numbers because the 32 bits can be rounded off to 24 bits. If a guard byte is not used, no rounding off is possible, and the effect would be the same as truncation (which can result in loss of accuracy very quickly, as will be shown later).

Numbers from 1.00 X 16^{-65} to F.FFFFF X 16^{+62} can be represented by this format, resulting in an approximate range of from 10^{-79} to 10^{+76} with an accuracy of six or seven decimal digits. Table 1 lists several decimal numbers along with their hexadecimal ANSI FORTRAN format equivalents.

The next format, shown in figure 3, is also a binary format and is implemented by Digital Equipment Corporation (DEC) and Hewlett-Packard in their BASIC interpreters. It consists of a 23 bit mantissa plus a "hidden bit," an 8 bit exponent and a sign bit.

This format assumes that the number is always normalized. Therefore, the most significant bit (MSB) of the mantissa is always one unless the entire number is zero. If the number is zero, (indicated by the special case of a 0 exponent) then the hidden bit is also zero. The sign bit is zero for a positive number and one for negative. Because all nonzero numbers have an MSB of one, it need not be explicitly represented in the format; hence only 23 bits in the mantissa.

The exponent represents a power of two in excess-128 notation, which is similar to excess-64 notation. The largest exponent, 2^{+127} is represented by the largest number, 1111 1111, and the smallest exponent, 2^{-127}, by the smallest nonzero number, 0000 0001. An exponent of zero (2^0) is represented by 1000 000, while the number zero is reserved to indicate a zero mantissa.

As in the case of the first binary format, a guard byte must be used during calculations so that round off is possible before returning from the floating point accumulator for storage in memory. In this format it is also necessary to explicitly represent the hidden bit during calculations. This is accomplished by expanding the 4 byte format

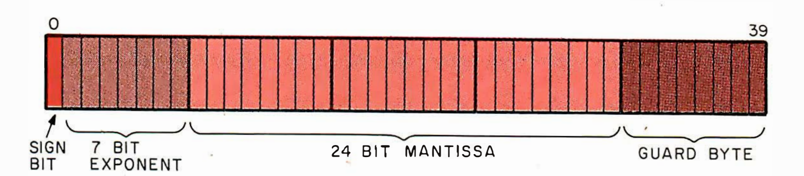


Figure 2: The ANSI FORTRAN floating point format showing the location of the "guard byte." The guard byte is an extra field which holds portions of intermediate calculations so that the final calculated value can be rounded off rather than truncated prior to further use.

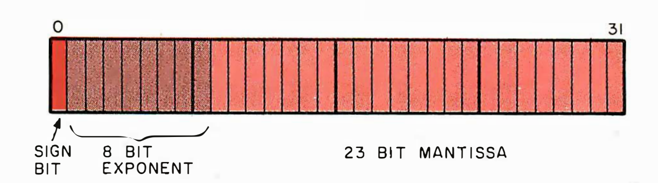


Figure 3: A binary floating point format used by Digital Equipment Corporation and Hewlett-Packard in their BASIC interpreters. It consists of a 23 bit mantissa with a "hidden" bit, an 8 bit exponent and a sign bit. The format assumes that the number to be represented is always normalized: the most significant bit of the number is always understood to be 1 unless the entire number is equal to 0. This assumed "1" bit is the so-called "hidden" bit.

Decimal Number	Hexadecimal Floating Point Number (Hexadecimal Digits)	
1.00	41	100000
6.00	41	600000
−1.00	C1	100000
0.50	40	800000
−0.50	C0	800000
100	42	640000
2^{16} (= 65,536)	45	100000
2^{-16}	3D	100000
-2^{-32}	B9	100000
0	00	000000
16^{-65}	00	100000
16^{+62}	7F	100000

Table 1: Several decimal numbers along with their ANSI FORTRAN floating point hexadecimal format equivalents (see figures 1 and 2).

Decimal Number	Binary Floating Point Number (Hexadecimal Digits)	
1.00	40	800000
6.00	41	C00000
−1.00	C0	800000
0.50	40	000000
−0.50	C0	000000
100	43	C80000
2^{16}=65,536	48	800000
2^{-16}	38	800000
-2^{-32}	B0	800000
0	00	000000
2^{-128}	00	800000
2^{+126}	7F	800000

Table 2: Examples of decimal numbers and their equivalents as encoded in the binary floating point format used in several BASIC interpreters (see figure 3).

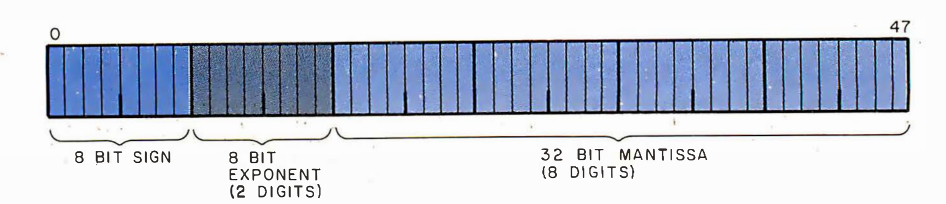


Figure 4: A BCD floating point format consisting of an 8 bit sign, an 8 bit exponent and a 32 bit (8 digit) mantissa.

into six bytes: one byte for the sign, one byte for the exponent, and four bytes for the mantissa (including the guard byte). As a result there is a fair amount of processing necessary to load and store the floating point registers.

This format has a range of from 2^{+126} to 2^{-128} or from approximately 10^{+38} to 10^{-38} with a 7 decimal digit accuracy (several examples are represented in table 2).

There are numerous BCD floating point formats currently in use. Mantissas range from as few as four digits to as many as 16 digits of accuracy, and exponents can typically range from 10^{+99} to 10^{-99}, or even 10^{+127} to 10^{-127}. The most popular format (see figure 4) has an 8 digit mantissa (four bytes of two digits per byte) with the decimal point assumed to be to the left of the most significant digit.

The mantissa sign is typically represented by a whole byte: 00 for positive and 0FF for negative. A variety of formats use one byte to represent the exponent.

One of the more frequently used formats is binary in the form of excess-128 notation. The exponent format itself is identical to the Digital Equipment Corporation format discussed earlier, but represents a power of ten instead of a power of two. Thus, an exponent of 84 base 16, using DEC's format, signifies two to the fourth power, and using the BCD format, ten to the fourth. The exponent represents the same power in both cases, but of different bases.

Eight digits are packed into four bytes in what is known as packed BCD (four bits represent one BCD digit).

The same format is usually used for both storage of data and actual calculations. This means neither a guard byte nor round off is used. The need for a guard byte is circumvented by using more significant digits than are actually necessary, eg: calculating to eight digits for 6 digit results, or calculating to nine digits for 8 digit results. This of course makes it necessary to use more memory per number for storage.

Some examples of numbers in this format are found in table 3.

Table 3: Several decimal numbers along with their equivalent floating point representations as encoded in BCD hexadecimal digits.

Decimal Number	BCD Representation (Hexadecimal Digits)		
1.00	00	81	10000000
6.00	00	81	60000000
−1.00	FF	81	10000000
0.50	00	80	50000000
−0.50	FF	80	50000000
100	00	83	10000000
2^{16}=65,536	00	85	65536000
2^{-16}	00	7C	15258789
-2^{-32}	FF	77	23283064
0	00	00	00000000
10^{+126}	00	FF	10000000
10^{-128}	00	01	10000000

Format Pros and Cons

Each of these basic floating point formats has its own particular advantages and disadvantages. Which format is best is dependent upon the requirements of the particular application: speed, small memory size, variable mantissa length, ease of coding in a given computer architecture, ease of interfacing to other software routines, etc.

The BCD format with its variations is by far the most popular in the personal computing field, probably because it is the easiest to program. The relative ease in converting from an ASCII representation of a number to the BCD format and back is a key factor, as is the ease with which the

Continued on page 180


Introducing the
SPACE BYTE™ 8085 CPU
A self-contained
computer

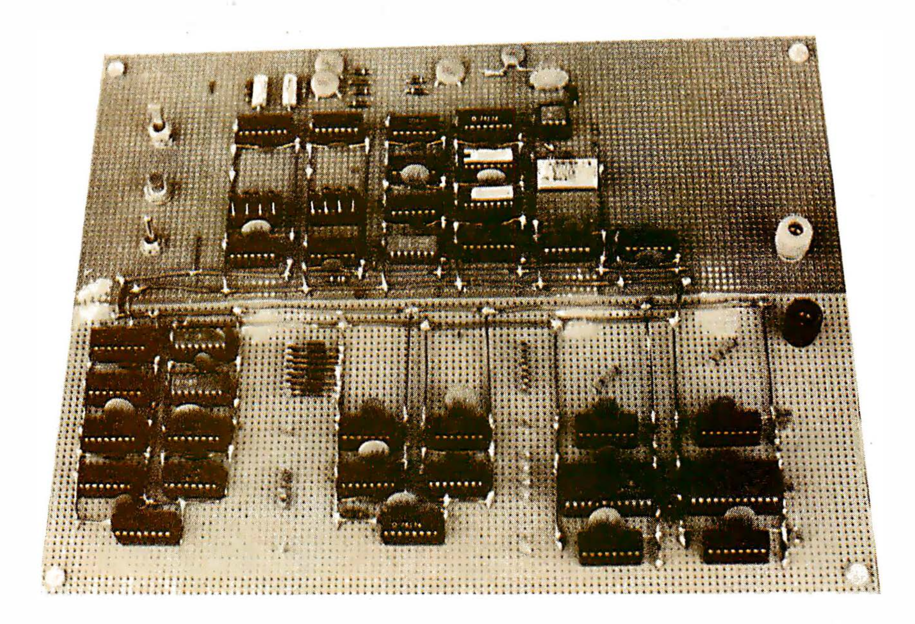


Photo 1: The author's computer, seen from the component side, was assembled using two sections of perforated board (0.1 inch grid) and sockets for all integrated circuits. The arithmetic unit is in the lower right hand section in this photograph, the eight memory circuits are in the lower left hand region, and the control section is implemented by the parts on the board at the top of this photograph.

Building a Computer from Scratch

Hilary D Jones
364 Princeton La
Danville CA 94526

With so many excellent microprocessors available today, the experimenter needs a good reason to design and build a personal computer from scratch. That reason will certainly not be one of economy. The best available microprocessors offer so much capability at such a low price that one cannot hope to save money by building a computer from scratch. For many, the reason will simply be the challenge of doing it. For others, the reason will be more practical (perhaps to gain some capability not readily available from an off-the-shelf microprocessor). And for still others, the reason will be to learn more about the techniques of computer design.

While any of these reasons is certainly valid, the design of a computer from the ground plane up is still generally regarded as an art that only the foolhardy would undertake. In reality, though, the job is not nearly as mysterious as it seems. For proof I offer the fact that when I began this project I had no design experience with TTL (or experience with any form of electronics design for that matter). Indeed, I chose this project to *learn* how to use TTL parts, on the assumption that the microprocessor I planned to buy would eventually become bored talking to my TV set.

Because of my inexperience with TTL circuitry, I chose to simplify the design as much as possible at every step. As a result, the major strengths of this computer are its low cost and its simplicity. With judicious shopping, it should be possible to construct the computer for around $65, including everything but the power supply. With only four instructions, the computer offers an instruction set that is guaranteed not to overwhelm the novice. At the same time, the signals that drive the various modules of the computer are readily accessible so that the electronics can be seen to work "as advertised."

Despite the simplicity of the computer, its microprogrammed bus oriented architecture conforms to the design principles in the most modern of minicomputers.

This article gives the groundwork from which a serious student or hacker can design and build his/her own personal computer.

Circle 144 on inquiry card.

Mnemonic	Object Code	Operation Performed
WIO N	00nnnnnn	Wait for input to location N. Display current contents of Location N while waiting.
ADD N	01nnnnnn	Add data in location N to accumulator.
STN N	10nnnnnn	Store negative of accumulator in location N.
JGE N	11nnnnnn	Jump to location N if accumulator is greater than or equal to zero.

Table 1: The instruction set for the computer. N is any 6 bit integer. The bits of N are denoted by nnnnnn.

At the same time it also describes a very simple computer, one that can be built by a student as a science project, by a teacher for a laboratory demonstration, or by a novice hacker who just wants to learn about computers without a large investment.

The Instruction Set

The most important task facing the computer designer is choosing the instruction set. In the case of this computer, every effort was made to choose the simplest possible instruction set. Therefore, multiple word instructions, stacks, register files, interrupts and elaborate IO facilities were not permitted. An 8 bit word length was chosen because it is the smallest size that can be reasonably expected to use one word per instruction. This constrained me to an instruction set of four op codes and a directly addressed memory space of 64 bytes. The instruction set is summarized in table 1.

The ADD instruction is included for obvious reasons. The STN instruction was chosen to store the negative value of the accumulator's contents so that both subtraction and addition could be done. (In particular, by executing STN N and ADD N in sequence, the accumulator can be cleared.)

The JGE instruction is an all purpose test or branch instruction. By clearing the accumulator before executing a JGE, an unconditional branch results. Alternatively, by placing a number in the accumulator, the JGE tests whether the number is positive or negative.

The WIO instruction provides the only means for loading and examining memory. There are no front panel switches for this function, so everything must be done under the control of a suitable program (including the loading of that program itself). Therefore, the WIO instruction requires special attention. When executed, WIO N brings the computer to a halt with the contents of location N displayed in the LED display. At this point, the user will enter data into a switch register. When the continue button is pressed, the data will be written into memory location N, thus destroying the data just displayed. In effect, the instruction combines the wait, input and output instructions of the conventional computer. A particularly useful application of this instruction occurs when the instruction at location N is a WIO N+1 instruction. At that point the data entered by the user becomes the next instruction to be executed!

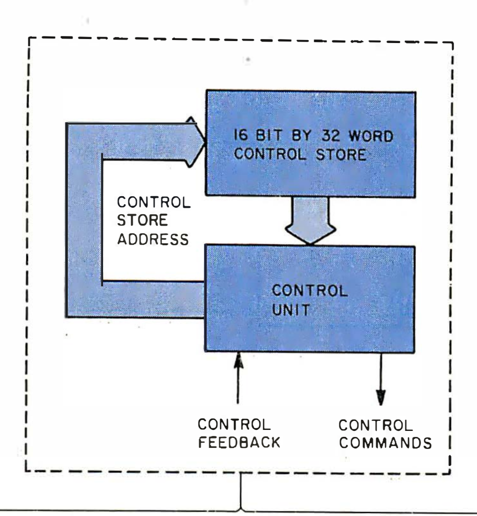
The most important program for this computer is the bootstrap program. Other programs are left to the reader to devise. The bootstrap program, which provides the simplest practical way to load data into the computer, is loaded as follows. When the reset button is pressed, the computer will wait to accept data into location 0. The data will be stored, then executed when the continue button is pressed. Needless to say, the data loaded must be chosen carefully if the user is to be able to keep control of the computer. The data that permits this is the WIO 1 instruction. When this instruction is loaded into location 0 and executed, the computer will halt, ready to accept data into location 1. When the continue button is again pressed, the new data is stored. The computer resumes execution with the next instruction in memory, namely, the instruction just entered into location 1. Again, that instruction must be carefully chosen: a WIO 2 is a good choice. The computer will again halt, at which time a JGE 0 should be entered. After JGE 0 is loaded into location 2 and executed, we will have completed entering the bootstrap program. The JGE 0 will unconditionally jump to the start of the bootstrap program (location 0) because the accumulator is cleared at restart time.

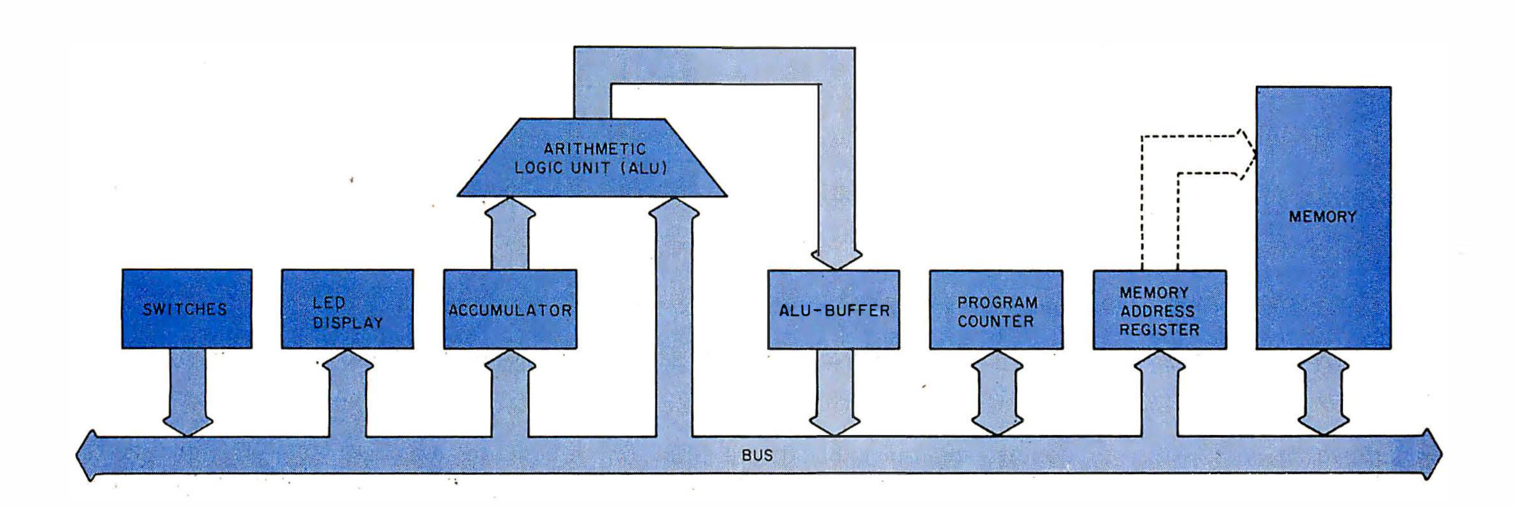
To use the bootstrap program, enter pairs of bytes as follows: a 6 bit address followed by eight bits of data to be placed at that address. For example, if, in response to the first two halts in the bootstrap program, we enter an octal 003 followed by an octal 010, then the value 10 will be placed in location 3. In this case, the bootstrap program returns to location zero, where it is ready to accept another pair of bytes. Once a program is loaded, we can execute it by entering the appropriate JGE instruction in response to the next halt instead of the address data pairs.

Hardware for the Computational Unit

The computer is shown in block diagram form in figure 1. The control unit, to be discussed later, interprets the instruction

Figure 1: *Block diagram of the author's computer, which is organized around an 8 bit bidirectional bus.*





set and generates the control signals which tell the computational unit how to execute instructions. This is a true microprogrammed computer with a 32 word by 16 bit control store. (For control store contents see table 2.) The system is organized around an 8 bit bidirectional bus. Because of this, each module that uses the bus may be built as an independent unit without regard to how the other modules work, an obvious advantage.

When two modules need to exchange data, one will put the data on the bus while the other will read it from the bus. The arrows in figure 1 show the directions in which such data transfers can be made. The only restriction is that no two modules are permitted to put data on the bus at the same time. The use of the bus system allowed me to build the entire computational unit before giving detailed thought to how the control unit would be implemented (the algorithm for successful computer design being "divide and conquer").

The arithmetic logic unit buffer register deserves comment. During an ADD operation, data from memory is placed on the bus. The arithmetic logic unit reads the data from the bus and adds it to the accumulator. The output of the arithmetic logic unit must eventually find its way back to the accumulator. This is done by putting the data on the bus, an action permitted only after the memory is no longer using the bus. The arithmetic logic unit buffer is provided to give a temporary holding place for the sum until the memory can release the bus.

Note that there is no instruction register or memory data register. This represents a departure from conventional computer design made possible by the simplicity of the instruction set. The conventional memory address register and program counter *are* present, however, and serve their usual purpose.

For even this simple computer, there are some 30-odd signals between the control and computational units. Therefore, I found it essential to establish a system for naming the signals. Names are best chosen to suggest what the signal does as well as the voltage

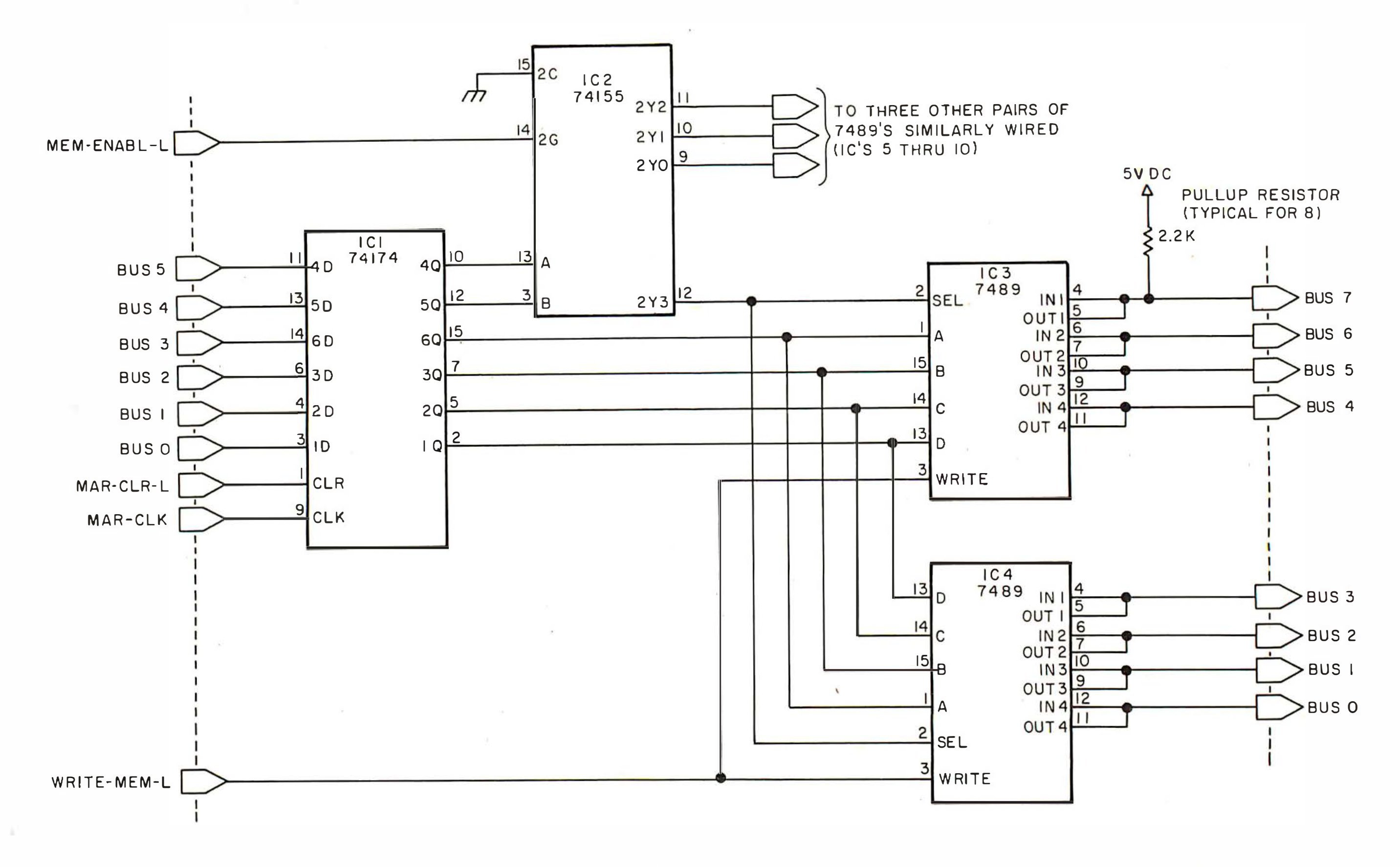


Figure 2: The memory address register and the memory. Data is addressed by six bits of the 74174 memory address register and stored in the 7489 memories, each of which has 16 4 bit registers. A total of eight 7489s are required to make up the 64 byte memory.

required to achieve the effect. For example, AC-CLR-L is a signal that, when brought low, clears the accumulator. Conversely, when PC-INCR-H is brought high, the program counter is incremented. The eight bus lines are named BUS0 thru BUS7 (in order of arithmetic significance). The master clock is named MCLK; its complement is called CCLK.

The memory and memory address register are shown in figure 2. I chose to base my system around the 7489 64 bit memory integrated circuit largely because I happened to have them. In a redesign, a 2101-based memory might be a slightly better choice, but the present design does have the advantage of showing how multiple chip memories are controlled. Each 7489 contains sixteen 4 bit registers; eight 7489s are required for a 64 byte memory. Data is addressed by a 6 bit memory address register (a 74174).

The memory address register is loaded with data on the bus by MAR-CLK. Alternatively, it can be cleared by MAR-CLR-L (eg: when the reset button is pressed). The two high bits of the memory address register are decoded together with MEM-ENABL-L by a 74155 decoder. If MEM-ENABL-L is low, the high two bits select one of four pairs of 7489s, and the low four bits select one of the 16 registers in the selected pair. (If MEM-ENABL-L is high, the memory is disabled.) In this way, a byte of memory is addressed. Now, if WRITE-MEM-L is low, that byte will be written using data from the bus. But when WRITE-MEM-L is high, the complement of the data at the addressed byte will be placed on the bus. The fact that the 7489 complements data stored in it is used to advantage by the STN instruction, as we will see later.

The accumulator (AC), arithmetic logic unit, and arithmetic logic unit buffer are shown in figure 3. The accumulator is constructed from a pair of 74175 integrated circuits. It is cleared by AC-CLR-L (eg: at restart time), and it is loaded by AC-CLK. The sign bit of the accumulator is sent (as AC-GE-L) to the control hardware, where it is used for the JGE instruction. The arithmetic logic unit, in the form of a pair of 74181s, is used in two ways. When executing an ADD instruction, ALU-ADD-L will be brought low, so that the arithmetic logic unit computes AC plus memory. This sum is then latched into the arithmetic logic unit buffer (a pair of 74173s). Once memory is no longer using the bus, BUS-BUF-L can be brought low to place the sum on the bus. The sum can then be latched back into the accumulator to complete the add cycle. Alternatively, to execute the STN instruction, ALU-ADD-L will be brought high. Now the 74181s will compute "accumulator minus one," which is latched into the buffer and eventually written into memory. The convenience of a complementing memory can now be appreciated, since in two's

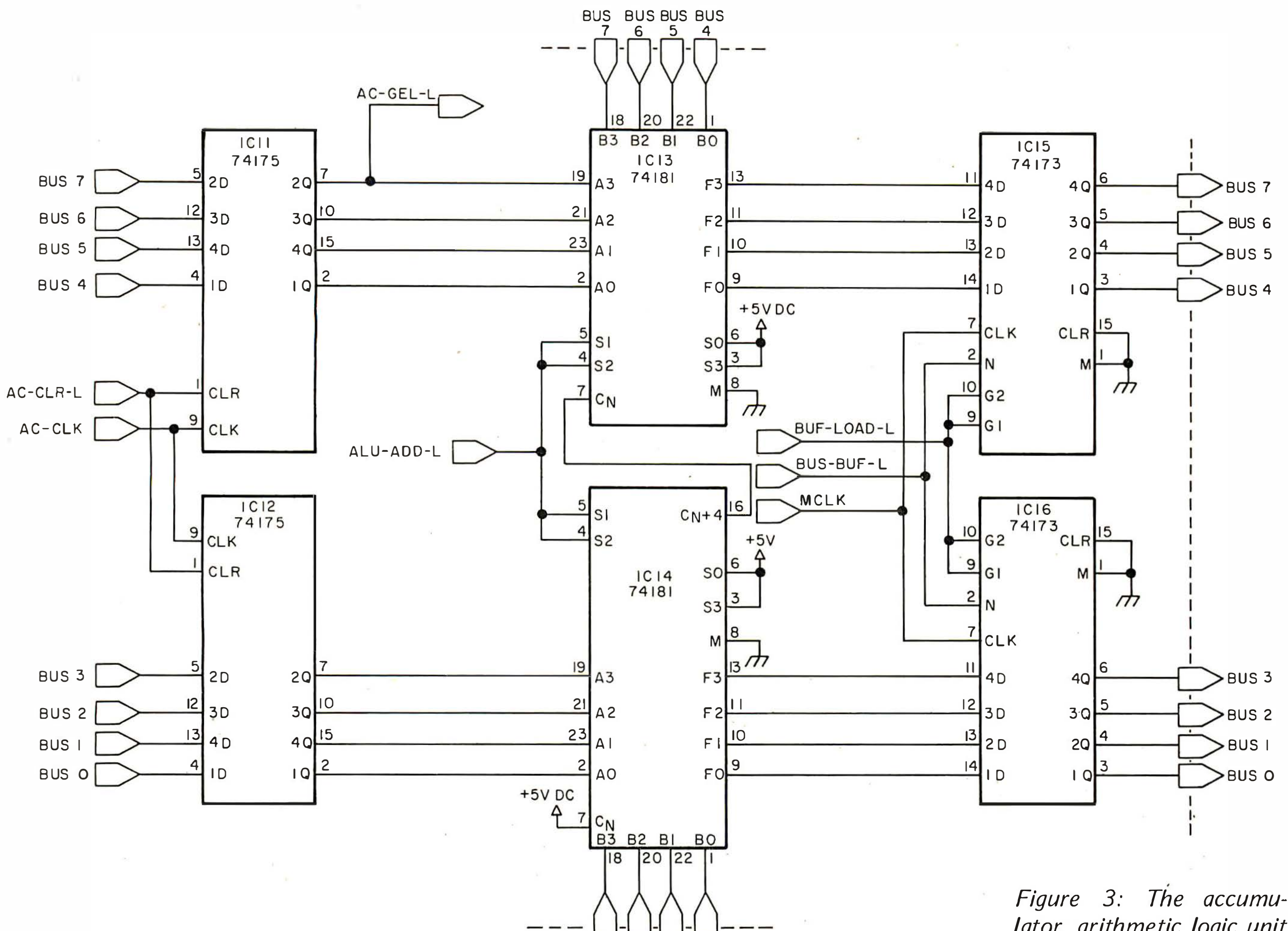


Figure 3: The accumulator, arithmetic logic unit and arithmetic logic unit buffer. The accumulator is made up of two 74175 quad D flip flops, while the arithmetic logic unit consists of two 74181 arithmetic units. Two 74173 integrated circuits form the arithmetic logic unit buffer.

complement arithmetic the complement of AC−1 is −AC. (With a 2101-based memory, the inversion would have to be done with extra hardware.)

The last part of the computational unit, shown in figure 4, consists of a program counter, LED display, and switch register. The program counter consists of two 74LS161 counters and two three state buffers. (The 74161 is not an acceptable substitute for the 74LS161 because of differences in the way their clocks behave, a fact I learned the hard way. The program counter is cleared by PC-CLR-L. It can be loaded (incremented) on the next clock transition after PC-LOAD-L (PC-INCR-H) becomes low (high). The output of the program counter is enabled onto the bus by PC-BUS-L.

The LEDs are driven by ordinary inverters. The inverters insure that the LEDs are lighted for the high bus lines rather than the low ones. The switch register (a DIP switch) is wired so that a closed switch drives the associated bus line low. This is because the memory complements data. The switch outputs are enabled onto the bus by a pair of 74125 three state buffers under the control of BUS-SWI-L.

Hardware for the Control Unit

The control unit, shown in figure 5, is responsible for providing the various signals in the proper sequence to drive the computational unit. To simplify the design, a microprogrammed architecture was chosen. In this design, the control logic is held in a pair of 74288 programmable read only memories in the form of a 12 word (16 bits per word) microprogram. When one or another word is selected from the programmable read only memory, the individual bits of the selected word are delivered more or less directly to the computational unit as individual signals. For example, the PC-INCR-H bit of the microprogram directly drives the PC-INCR-H line to the program counter. Similarly, the BUF-LOAD-L bit directly drives the BUF-LOAD-L line to the arithmetic logic unit buffer.

Several other lines can be readily identified that are directly driven by the programmable read only memory. From this it is clear that the problem of designing a control unit reduces to deciding which



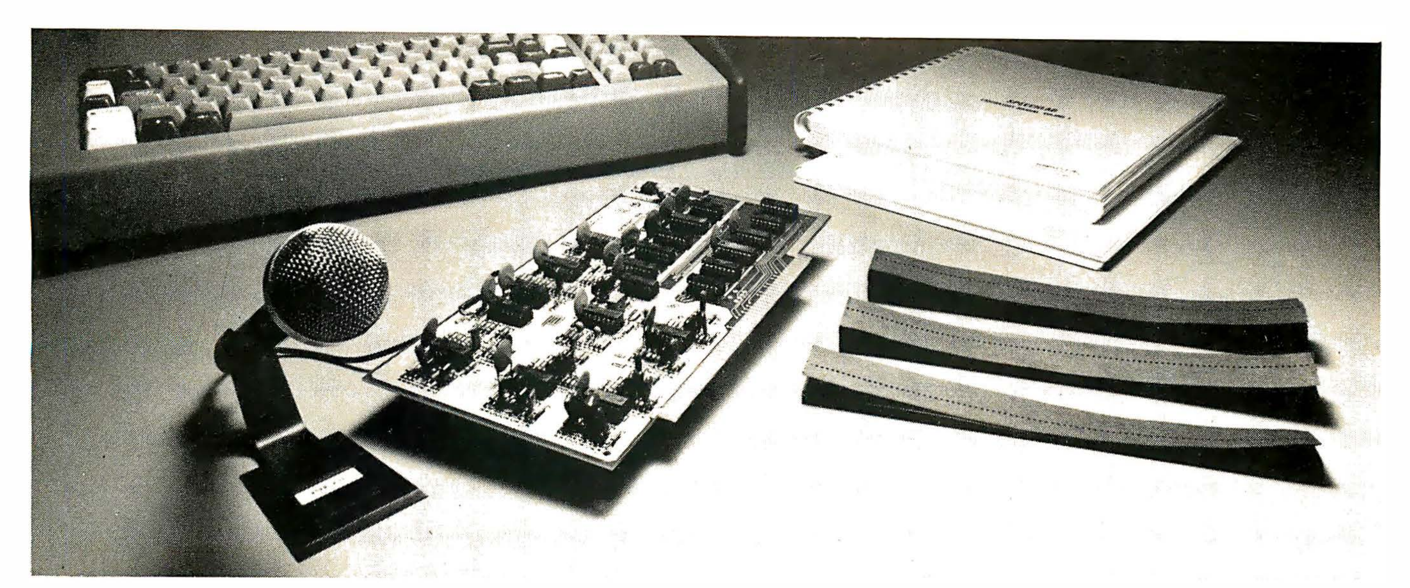
Figure 4: Program counter circuitry.

signals are to be high or low at what time, programming a programmable read only memory to contain this information, and devising a way to select the proper word from the programmable read only memory at the proper time so that the appropriate signals can be generated.

Occasionally, microprogram bits do not drive signal lines directly but must first undergo some transformation. For example, in figure 5 we see that the AC-LOAD-L bit of the microprogram is gated with CCLK to create the clock that loads the accumulator. (It would not be permissible to drive the accumulator directly from AC-LOAD-L, because the signal has to be delayed by a half clock cycle.) Similarly, MAR-CLK is derived by gating MAR-LOAD-L together with CCLK, and PC-LOAD-L is created by gating together AC-TEST-L and AC-GE-L, so that the program counter is loaded only when the microprogram allows it and the accumulator is not negative.

The BUSDAT0 and BUSDAT1 lines are another case in which a transformation is required. In this case the two lines are decoded by a 74155 decoder to select one of four possible sources of data for the bus, namely, the switch register, program counter, ALU-BUFFER and memory. These are selected by BUSDAT1, BUSDAT0 values of (L, L), (L, H), (H, L) and (H, H), respectively. This arrangement saves bits in the microprogram as well as insuring that only one device can put data on the bus at any one time. Note that the open collector memory used is logically connected to the bus by enabling an appropriate memory chip. (The memory chips must also be enabled before writing memory.)

In order for the microprogram to deliver its control signals to the computational unit in the appropriate order, some means must be provided for sequencing thru the words in the programmable read only memory. In this computer, we have allocated six bits (BASE0 thru BASE4, and OPJMP-H) to accomplish this. First, assume that OPJMP-H is low. "BASE" then determines a microprogram address (the base address) which is fed forward directly to the 74174 microprogram address register. When CCLK goes high, the



Talk to your computer for $299 with SpeechLab.™

Use SpeechLab to directly control any S-100 Bus Computer such as Sol, IMSAI, Altair and so on. SpeechLab can teach you almost as much as the Bell Laboratories know about voice recognition, voice control and computer input.

SpeechLab digitizes and extracts data from speech wave form and applies pattern matching techniques to recognize the vocal input. Response is real time. The system features 64 bytes of storage per spoken word and can handle up to a 64 word vocabulary. And recognition after very little practice is 95 percent or better.

$299* assembled and tested

When we talk price everybody's skeptical. And why not? We give you a complete hardware/software system, a 275 page laboratory manual, 95 page hardware manual and high fidelity microphone.

The lab manual includes 35 graded experiments with over 100 tables and graphs. In fact, it's the only introductory volume on speech recognition currently available.

Software includes SpeechBasic Basic programming language in source and paper tape, assembly language speech recognition program in source and paper tape, hardware self-test program in source and paper tape. SpeechBasic plot, correlation, recognition and advanced recognition programs are offered in source.

Hard to believe, you bet. True? A Los Angeles customer says, "I love your kit!!! I have 40 boards and 2 IMSAI's and your kit was the best documented of them all. I love the way you integrated the software and hardware together. I love your lab manual."

We loved those comments. They tell the story better than we ever could. The LA customer did ask who the founders of the firm were so he could relate better. They're a couple of gifted young engineers who got tired of the big firm, big technology trip and decided to take a chance with a better idea.

You can't get better quality
You can't get more performance

Sure, more complex, higher price equipment is available for about 50 times more money. It won't do much more than you can do with SpeechLab. And the quality and state-of-the-art engineering can't be any better. We use CMOS design for low power and ultimate reliability.

See SpeechLab at your nearby computer store

Selected computer stores have SpeechLab on display. Visit your nearest. If he doesn't have it, ask him to contact us or simply write us directly.

*Available in kit form for slightly less.

Heuristics, Inc.
Box B, 900 N. San Antonio Rd.
Los Altos, CA 94022, Phone (415) 948-2542

☐ Send me SpeechLab. I enclose $299. California residents add sales tax.
☐ Send me more information.

☐ Master Charge
☐ BankAmericard (Visa)
Acct. No.______
Date Card Expires______
Interbank No.______
(Master Charge only)

Name ______
Street ______
City______ State______ Zip______

In Canada: Trintronics Ltd.
186 Queen St. W.
Toronto, Canada M5V 1Z1
Telex 06-22634

Heuristics INC.

Circle 65 on inquiry card.

Figure 5: The microprogrammed control unit of this computer.

SOFTWARE = TSC

TEXT EDITING SYSTEM: This 6800 editor is unlike any other micro editor. As well as the usual features, it also includes; content oriented commands, local and global commands, block move and copy, append and overlay features, as well as very comprehensive string manipulators. 5K

SL68-24	MANUAL & SOURCE LISTING	$23.50
CT68-7	OPTIONAL CASSETTE	$6.95
PT68-6	OPTIONAL PAPER TAPE	$8.00

TEXT PROCESSING SYSTEM: A great companion to the TSC editor. The processor will allow convenient paragraphing, right hand justification, paging, titling, and general text formating. 4K

SL68-29	MANUAL & SOURCE LISTING	$32.00
CT68-9	OPTIONAL CASSETTE	$6.95

TSC MNEMONIC ASSEMBLER: Another 6800 resident assembler? Yes! But this one is many times faster than others due to a very efficient symbol handler. All the standard motorola options included as well as psuedo ops. It is very modular making it quite easy to adapt to most systems. 5K

SL68-26	MANUAL & SOURCE LISTING	$23.50
CT68-8	OPTIONAL CASSETTE	$6.95
PT68-9	OPTIONAL PAPER TAPE	$8.00

DISASSEMBLER FOR 6800: Now one that is reasonably priced and includes the source listing!

SL68-27	MANUAL & SOURCE LISTING	$9.00
PT68-7	OPTIONAL PAPER TAPE	$4.00

TSC 6800 ARITHMETIC ROUTINES

SOAP: A very fast, 4 byte binary, floating point package. Includes integer and conversion routines as well.

SL68-25 $10.00

FLOATING POINT PACKAGE: A BCD math package with 9 digits of precision.

SL68-4 $6.50

SCIENTIFIC FUNCTIONS: Requires SL68-4 and provides all scientific functions including SIN, TAN, LOG, LN, HYPSIN, and others.

SL68-20 $10.00

SPACE VOYAGE: A full blown Star Trek program written in 8080 and 6800 assembler language. Runs much faster than similar BASIC versions and requires about 1/5 of the memory! Each game is different as you try to save the federation using your phasers, photon torpedoes, and shields. Searching out the menacing Klingons is accomplished using the short and long range scanners as well as moving about by firing the warp engines. Beware of sudden attacks, space storms, supernovas, and other unexpected events. This game is very addicting! 4K

SL68-5	6800 SOURCE LISTING	$12.00
CT68-1	OPTIONAL CASSETTE	$6.95
SL80-9	8080 SOURCE LISTING	$12.00
PT80-1	OPTIONAL PAPER TAPE	$7.00

DIAGNOSTICS FOR 6800: No system is complete without a set of diagnostic programs. Includes 5 memory tests, serial I/O tests, parallel I/O tests, plus others.

SL68-23 SOURCE LISTING $10.00

MICRO BASIC PLUS: The best small BASIC available for 6800. In just 3¼ K, a complete interpreter including GOSUB, IF THEN, FOR NEXT, DIM, ON GOTO and GOSUB, READ DATA, plus the functions RND, SPC, TAB, EXP, and ABS. Five full digit integer math is supported.

SL68-19	MANUAL & SOURCE LISTING	$15.95
CT68-5	OPTIONAL CASSETTE	$6.95
PT68-5	OPTIONAL PAPER TAPE	$6.00

6800 MULTI-USER SYSTEM!

Now you can have 4 simultaneous users, all running BASIC, and independently! Give your micro the power of a large mainframe. Applications include: a great educational tool, small business applications such as multi station inventory system, industrial computer power expanding the personal computer system,. . .etc. The system is presently available only for the SWTPC 6800 computer system and will support cassettes, floppy discs, and a printer. For complete details of the multi-user system, write for our 4 page brochure.

MUB-68 $129.95

ALL SOFTWARE CONTAINS: Complete commented source listing, users manual with complete instructions, printed hex dump, sorted symbol table, and sample output.

PROGRAM-OF-THE-MONTH CLUB.™ $2.00 for a one year membership. No obligations!

HOW TO ORDER: All orders should include check or money order. Add 3% for postage and $1.00 for handling for orders under $10.00. Send 25¢ for complete TSC software catalog.

TECHNICAL SYSTEMS CONSULTANTS, INC.

Box 2574 W. Lafayette, IN. 47906

Circle 146 on inquiry card.

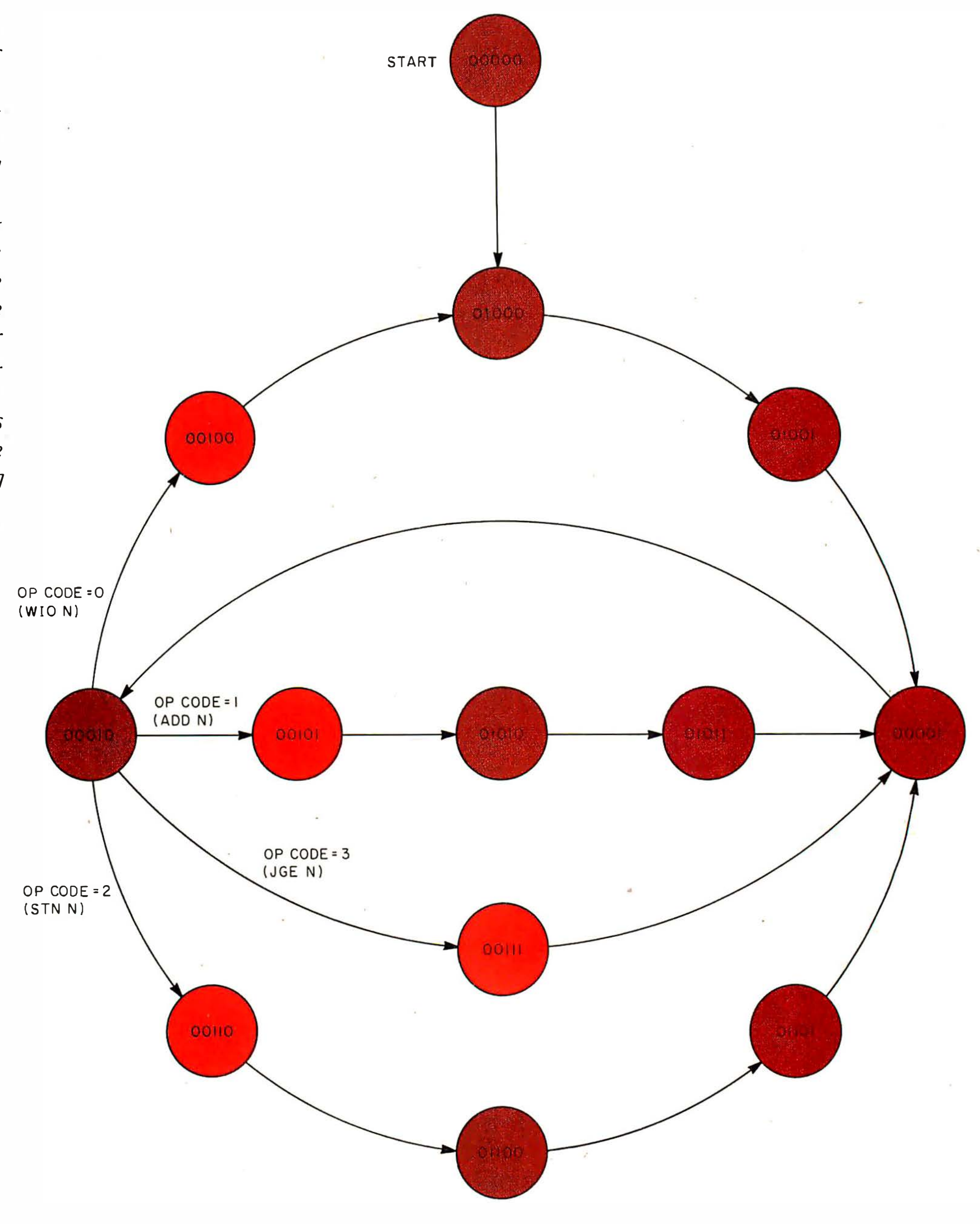


Figure 6: A finite state graph representation of the microcode shown in table 2. The five digit binary numbers shown in each state are the control memory addresses when the computer is in that state. Note that a 4 way branch occurs at state 00010 (left side of the graph), indicating the four different op codes implemented for this computer. The resulting initial states for the four op codes are shown in contrasting color.

base address becomes the new microprogram address, thus defining the next microprogram word to supply control signals. Continued clocking of the 74174 thus causes the microprogram to sequence thru whatever steps it has chosen for itself, and at the same time, to deliver control signals to the computational unit.

To this next address scheme we must add some means of varying the microprogram flow based on the op codes encountered. This is the reason for having the OPJMP-H bit. When it is high, the base address is no longer the next address. The latter is formed by performing a logical OR of the base address with the op code (assuming that the bus holds the instruction to be executed). Ordinarily OPJMP-H will be set high in a microinstruction that has the two lowest bits in BASE set to zero. In that case, each op code will produce a different next address.

To see the next address scheme in action, consider table 2 and figure 6 in which the microcode for the computer is shown. If we start at microcode address 00000, which is the case when the restart button is pressed, then we find that OPJMP-H is low, so the next address will be at BASE=01000. Subsequent addresses are 01001, 00001, and finally 00010. At this point, OPJMP-H goes high. Let us assume for sake of example that an ADD instruction has been placed on the bus (op code = 01). Then the next address will be 00100 v 01 = 00101. The subsequent addresses are then 01010, 01011, 00001,

Control Source Address (Binary)	Programmable Read Only Memory A								Programmable Read Only Memory B							
	BASE 4	BASE 3	BASE 2	BASE 1	BASE 0	OPJMP-H	ALU-ADD-L	WAIT-L	PC-INCR-H	BUF-LOAD-L	AC-TEST-L	BUSDAT 1	BUSDAT 0	MAR-LOAD-L	AC-LOAD-L	WRITE-MEM-L
0 0 0 0 0	0	1	0	0	0	0	*	1	0	1	1	*	*	1	1	1
0 0 0 0 1	0	0	0	1	0	0	*	1	0	1	1	0	1	0	1	1
0 0 0 1 0	0	0	1	0	0	1	*	1	1	1	1	1	1	1	1	1
0 0 1 0 0	0	1	0	0	0	0	*	1	0	1	1	1	1	0	1	1
0 0 1 0 1	0	1	0	1	0	0	*	1	0	1	1	1	1	0	1	1
0 0 1 1 0	0	1	1	0	0	0	*	1	0	1	1	1	1	0	1	1
0 0 1 1 1	0	0	0	0	1	0	*	1	0	1	0	1	1	1	1	1
0 1 0 0 0	0	1	0	0	1	0	*	0	0	1	1	1	1	1	1	1
0 1 0 0 1	0	0	0	0	1	0	*	1	0	1	1	0	0	1	1	0
0 1 0 1 0	0	1	0	1	1	0	0	1	0	0	1	1	1	1	1	1
0 1 0 1 1	0	0	0	0	1	0	*	1	0	1	1	1	0	1	0	1
0 1 1 0 0	0	1	1	0	1	0	1	1	0	0	1	*	*	1	1	1
0 1 1 0 1	0	0	0	0	1	0	*	1	0	1	1	1	0	1	1	0

OPJMP-H: Enables or disables control store address modification.

BASE 1 – BASE 0: Low order address of next program step (modified by instruction code of user program through IC33 and IC34).

BASE 4 – BASE 2: High order address of next program step.

Note: "*" Indicates a "don't care" bit.

Table 2: Standard microcode to implement the instruction set listed in table 1. Six bits of each word (BASE 0 thru BASE 4 and OPJMP-H) have been reserved to tell the computer where the next word in the program sequence is located. For instance, upon startup, the computer is at control source address 00000. BASE 0 thru BASE 4 (BASE 4 is the most significant bit) have the values 01000, indicating that the computer is to go next to control source address 01000, and so on. When control source address 00010 is reached, OPJMP-H is set equal to 1. At this point one of the four possible instructions will be executed depending on the values of the op code on bus lines 6 and 7 (see figures 5 and 6).

and so on. Note that address 00001 marks the beginning of an infinite microprogram loop that fetches, interprets and executes the instruction set.

One more microprogram bit (WAIT-L) remains to be described. It provides a way to halt our computer at the WIO instruction. As long as this bit remains high, the rather involved network of NAND gates conspires to pass clock pulses so that the microprogram executes freely. But when the WAIT-L bit goes low, a flip flop changes state and permits the clock synchronization network to turn off the clock at the end of the current clock cycle. This stops the computer, but it can still be single stepped by hand so that we can study the microprogram. Clock pulses will remain disabled until the continue button is pressed; then the flip flop reverts to its original state, enabling the clock synchronization network to pass pulses at the start of the next clock cycle. This restarts the microprogram, and hence the computer. The edge trigger on the continue button prevents the computer from continuing right thru several WIO instructions. (A properly debounced continue button is essential here for the same reason.)

The restart button shown in figure 5 allows us to prime the computer or to abort a malfunctioning program. Pressing it causes the accumulator, program counter, memory address register and microprogram address register to be cleared and paves the way for the bootstrap program to be reloaded.

In table 2 I have listed the version of microcode that implements the instruction set laid out earlier. The various fields are identified at the top of the table, with additional explanatory notes at the bottom. Figure 6 uses the information in table 2 to show a complete state diagram of the instruction set. Each state (a colored circle) is one address in the control memory represented by one line in table 2.

One point worth keeping in mind concerns timing. At the beginning of each cycle,

Table 3: Power wiring table for the circuits in figures 2 thru 5.

Number	Type	+5 VDC	Gnd
IC1	74174	16	8
IC2	74155	16	8
IC3	7489	16	8
IC4	7489	16	8
IC5	7489	16	8
IC6	7489	16	8
IC7	7489	16	8
IC8	7489	16	8
IC9	7489	16	8
IC10	7489	16	8
IC11	74175	16	8
IC12	74175	16	8
IC13	74181	24	12
IC14	74181	24	12
IC15	74173	16	8
IC16	74173	16	8
IC17	74LS161	16	8
IC18	74LS161	16	8
IC19	74125	14	7
IC20	74125	14	7
IC21	7404	14	7
IC22	7404	14	7
IC23	74125	14	7
IC24	74125	14	7
IC25	74288	16	8
IC26	74288	16	8
IC27	74174	16	8
IC28	74155	16	8
IC29	555	8	1
IC30	7400	14	7
IC31	7400	14	7
IC32	7402	14	7
IC33	7432	14	7
IC34	7408	14	7
IC35	7410	14	7

as CCLK goes high, a new microinstruction will appear at the output of the programmable read only memories. This might cause data to be placed on the bus, or a sum to be formed by the arithmetic logic unit, etc. In any case, by the middle of the clock cycle, when CCLK goes low, it is assumed that all such data has settled. Therefore it will be safe to latch the data set up during the first half of the cycle into some appropriate device. By the end of the cycle, the latched data will also be stable, so that a new microinstruction can be safely executed. In this way we have avoided timing problems without going to a two phase clock. I estimate that the cycle time of the computer could approach 300 ns, although I have not pushed the computer to its limit.

Summing Up

The design of this simple computer will certainly not appeal to everybody. Expanding the design to a 12 bit word length would permit much more flexibility in the instruction set, perhaps enough to even make the computer practical. For example, indirect addressing might be thrown in, or a subroutine calling mechanism. The WIO instruction could be broken down into separate wait, input and output instructions, allowing the computer to do things like flash its lights. (The present design comes to a grinding halt with each flash.) A more elaborate bus structure, and some sort of flexible IO facility are other obvious improvements that one could try. Alternatively, a 12 bit word length could be used to increase the address space. Each of these changes would add to the cost and complexity of the design, but could lead to a more useful computer.

Although the requirements of the bootstrap loader do impose some severe constraints on the instruction sets that can be implemented with this architecture, you will probably want to try a few variations. It might be possible to implement two instruction sets: one defined by the lower 16 words of the programmable read only memory for loading programs, and the other in the upper 16 words for experimentation. A switch would be used to select between the instruction sets.

Once you have mastered the ideas behind the design of this computer, you'll be well on your way to building a serious computer. All you need to do is sit down and write out an instruction set that best fits your personal needs and then implement it in hardware. Bit slice microprocessors such as the AM2900 series offer a very attractive way of doing this. ■

SEE DEALER LIST ON PAGE 165.



The most cost effective products for your microcomputer.

RM64
64K bytes

THE EXTENSYS RM64 MEMORY BOARD provides the most cost effective system memory found in the industry. The RM64 provides this because of our low cost per byte when compared to our competition plus the increased reliability of a single board over multiple boards containing less memory. The board is S-100 bus compatible making it usable in over a dozen different microcomputer systems including ALTAIR and IMSAI. The RM64 is available in three configurations: 32K, 48K or 64K bytes of memory all on ONE board. The board is completely assembled, checked out and burned in for at least 50 hours prior to shipment. This complete testing procedure allows Extensys to provide a one year warranty on parts, labor and materials (assuming no misuse of the board occurs).

On board hardware is provided for:

- **Individual memory bank address selection in 8K byte increments;**
- **Complete dynamic refresh logic without loss of processing efficiency while programs are running;**
- **Board select logic which allows more than one 64K byte board per system;**
- **S-100 bus compatibility including on-board voltage regulator;**
- **Memory overlap which allows memory sharing the same address space to coexist in the same system;**
- **Write protection in 16K blocks; and**
- **Fully socketed for 64K, allowing 32K and 48K versions to be upgraded at a later date.**

Delivery of the RM64 is 15 to 30 DAYS upon receipt of order. Prices for the RM64 include shipping and handling prepaid in the continental United States.

EXTENSYS Corporation is also announcing several other new highly cost effective products. These include a total floppy disk system based around File I/O board and a multiprocessor operating system. The other product, which interfaces with the RM64 memory board to create a megabyte or more of memory and adds full DMA capability to the File I/O board, is a Board Select/DMA board. Both of these products are S-100 compatible.

Circle 58 on inquiry card.

Contact your local computer store or order directly from EXTENSYS

B-5

Please place my order for the following:

QTY	DESCRIPTION	AMOUNT
___	RM64-32K byte board @ $ 895 ea.	___
___	RM64-48K byte board @ $1195 ea.	___
___	RM64-64K byte board @ $1495 ea.	___
	Subtotal	___
	California residents add 6½% tax	___
	TOTAL	___

Shipping and handling prepaid in continental United States

NAME ____________

ADDRESS ____________

CITY ________ STATE ______ ZIP ______ SIGNATURE: ____________

PHONE (INCLUDE AREA CODE) ____________

extensys corporation
592 Weddell Drive
Sunnyvale, California 94086
(408) 734-1525

Please check method of payment:
- ___ Check Enclosed
- ___ BankAmericard No. ________ expiration date ________
- ___ Master Charge No. ________ expiration date ________

A 6502 Personal System Design:

Documenting Kompuutar — A Guide to the Details

The design information included with this article, together with the excellent documentation provided by MOS Technology on the 6502 design, should be complete enough to enable the advanced experimenter to build a similar Kompuutar. The details provided here cover a basic processor, but do not include a detail design of a programmable memory board which is a necessary part of a usable system. David Brader is currently working on an 8 K dynamic memory board, with invisible refresh, to be used in Kompuutar. The details of the wiring and construction of Kompuutar are shown in the several figures, tables and photographs, as well as listing 1. As a short guide to these materials here is a detailed table of contents to the article.

David Brader
POB 483
Electric City WA 99123

Front Panel Assembly: *This is the circuit with various displays and switches, which is mounted on the front panel, and talks to the front panel interface module via a multiconductor cable from P2 to J2.*

Front Panel Interface Module: *This is the logical interface between the processor's backplane bus and the front panel. It is the home of address decoding and the read only memory with the front panel service programs.*

Central Processing Module: *This is the heart of the Kompuutar system, a board which contains the 6502 processor, and associated buffering and clocking circuitry which defines the backplane bus structure of the system.*

TIM Interface Module: *This card is provided so that the MOS Technology "Terminal Interface Monitor," or TIM program, can be used with Kompuutar.*

Miscellaneous Items:

Kompuutar

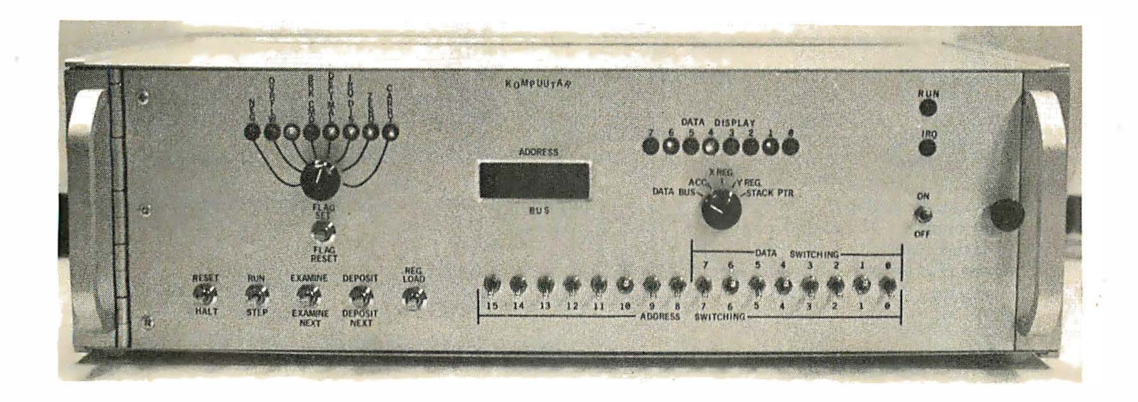


Photo 1: The completed Kompuutar, viewed towards its front panel. The controls of the front panel are chosen with the Data General NOVA's front panel as a mental model. In addition to the binary data display, there is a 4 digit hexadecimal address display (black rectangle) and an 8 bit binary flag display. The control panel is serviced by a read only memory routine.

Kaveat Kompuutar

It is with some trepidation that we present the details of the Kompuutar design. The design is complete and comprehensive, but Murphy is addicted to complete and comprehensive designs. Thus we'd like readers to be aware that there is a nonzero probability that errors exist in this magazine representation of author David Brader's design. We suggest that serious homebrewers of Kompuutar treat these pages as a detailed design guide, to be used with the standard design documentation of the chips involved. But as with any road map, do not be afraid to question and verify what you see with your own knowledge and experience.

David Brader reports that a local friend of his has built a second Kompuutar from the same set of blueprints which were the source of the circuit in this article. The experiences of the second builder were reflected in his corrections and changes to the drawings which are part of the normal "author proof" cycle applied to articles. Based on our own experiences with microprocessors, this report from David, and a tremendous amount of "desk debugging" of the article, we believe the information presented here is complete and buildable. However we highly recommend that readers who attempt to duplicate the design have sufficient experience with digital hardware and logic so that detailed understanding of its operation is possible. This is not a novice's project.

It all started at WESCON 1975, in San Francisco. It was there that I discovered what a "hospitality suite" is. In a hotel not far from the convention site, MOS Technology Inc had set up their WESCON hospitality suite. A hospitality suite is a bit like Las Vegas: some refreshments, a couple of elegantly decorative ladies, flashing lights and shiny gizmos, and the age old desire to persuade you and your money to part company.

I decided to stop and at least get a free drink. A man by the bar said, "Help yourself," so, being afraid of a one drink limit, I poured a double. As I left the bar area, I spotted a friend. We struck up a conversation about common friends and assignments, which lasted through half my drink and all of my clearheadedness. As our conversation ended, I noted some blinking LEDs and shiny new printed circuit boards. These boards were surrounded by several professional looking guests, giving the hardware an illusion of significance. So I went over to investigate.

I listened, wide eyed, to the saga of the MCS6502 as I slowly finished my drink. After the story ended, everyone seemed to be forming a line in a different part of the suite. Feeling part of the group now, I moved to the line. A little bit later, I remember being at the head of the line and the last thing I recall was handing two 20 dollar bills to a very pretty lady.

That evening, after sobering up, I discovered what I had done. There on my bed, stark naked, was a bright new MOS Technology Inc MCS6502 microprocessor chip and its manuals. Well, now the only

Backplane (P1) Pin Designation	Logic Diagram Mnemonics	Description	Backplane Pin Designation	Logic Diagram Mnemonics	Description
A	+ 5 V	voltage supply	1	GND	ground
B	IRQ 1	interrupt 1	2	IRQ 5	interrupt 5
C	A 0		3	A 1	
D	A 2		4	A 3	
E	A 4		5	A 5	
F	A 6	address bus lines	6	A 7	address bus lines
H	A 8	(even)	7	A 9	(odd)
J	A 10		8	A 11	
K	A 12		9	A 13	
L	A 14		10	A 15	
M	IRQ 2	interrupt 2	11	IRQ 6	interrupt 6
N	IRQ 3	interrupt 3	12	IRQ X	any interrupt pending
P	D 0		13	D 1	
R	D 2	data bus lines	14	D 3	data bus lines
S	D 4	(even)	15	D 5	(odd)
T	D 6		16	D 7	
U	SO	set overflow flag	17	SYNC	synchronize
V	PHICLK	Φ1 clock	18	PH2CLK	Φ2 clock
W	MASRST	master reset	19	RDY	ready
X	R/W	read and write	20	PANRST	panel reset
Y	IRQ 4	interrupt 4	21	NMI	nonmaskable interrupt
Z	GND	ground	22	+ 5 V	voltage supply

Table 1: Kompuutar bus list. This table gives the backplane socket pin identifications, mnemonics used in the logic diagrams, and a short description of the line's use. The pin designations are the standard ones printed on the Vector prototyping cards and embossed in the typical 44 pin sockets.

thing to do was to build a computer with the chip. After several days reading, I realized that building a computer was not going to be all that easy. I also realized that the initial $36.75 investment was but a drop in the proverbial bucket of costs.

Designing the Kompuutar System

Since my $36.75 investment was going to need considerable financial and design support, it was clear that making a project out of the computer would require planning. The first thing I had to accomplish was a specification of the features I wanted in my machine. I had had a good deal of experience with the Data General NOVA 1200 minicomputer, which led me to favor its functional front panel switch setup. With this input, I decided that the new machine would have the front panel functions of master reset, halt, program run, single instruction step, memory examine, examine the next memory location, deposit, desposit to the next memory location, load processor register, and enter data or address information from switches. I also knew that I wanted to be able to display the information on the data lines and address lines. I decided to use hexadecimal LED displays for the address bus information, but not for the data bus. My reasoning was that the address bus is always considered to be a numerical value, whereas the data bus is sometimes considered to be numeric data, but is sometimes viewed as a combination of individual bits. (If the data bus was showing hexadecimal E6 and you wanted to know if bit 5 was on or off, you would probably have to think for a while to make sure.) Another argument in favor of discrete LED indicators for each bit is the fact that hexadecimal displays are a bit more expensive.

After reading more about the MCS6502, a trait common to the other single chip processors revealed itself. The status register, accumulator, index register X, index register Y and stack pointer register do not come out of the chip on their own sets of pins. All that information was going to be hidden from the operator (me) sitting in front of the machine. I knew I would have to design digital logic to get that information out of the chip and displayed upon some sort of front panel. I even decided to go one step further and build in the capability to set or reset the status flags from the front panel. Being able to throw a switch and set the carry flag, for example, is a very handy capability when debugging a conditional branch in some program.

I decided to use toggle switches for the 16 data inputs because the state of individual switches could then be tested *in* software and used to control options in a program. This complicates the entry of an address (which is displayed in hexadecimal) but gains an ability to write applications programs which can be modified by the state of these input switches.

With these considerations in mind, the front panel design was firmed up as a starting point for the processor. I then started to work on the detailed logic design of what came to be called Kompuutar in my lexicon. After a month's work, I realized that the

Up your organization.

The Basic Box (left) and the Peripheral Plate.

With no-nonsense organizers from the Digital Group.

Not so long ago, the microcomputer domain belonged to a special group of creative, inventive folks — the inveterate hardware hackers who delighted in making a thing work and didn't really care all that much about how it looked.

The Digital Group was a part of it. Our original microprocessor systems were designed not to require any cabinets at all — they simply worked well.

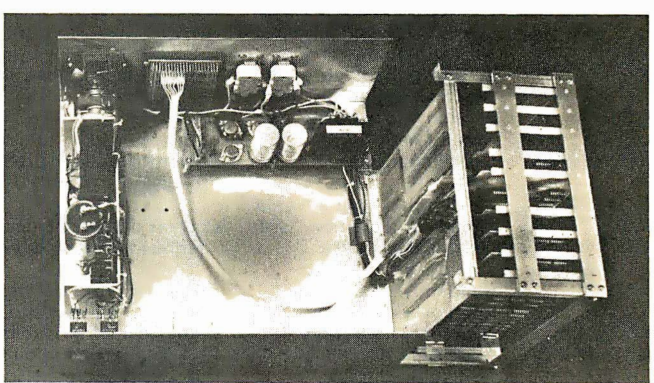
Of course, along the line we couldn't resist making a good thing look good too . . . and we added our complete line of custom, deluxe cabinets to cover up.

Well, we haven't forgotten those no-nonsense computer builders who just want a way to organize their systems. So the Digital Group has taken a step back to come up with a basic answer: The Organizers — the *Basic Box* and the *Peripheral Plate.*

Beautifully simple. No paint. No anodizing. No frills. Just exactly what you need.

The Basic Box houses your CPU, power supplies, fan, switches and I/O connectors in one tight little 16" by 17" package. It's available completely naked — a metal box with a card rack so you can add to it from your own parts supply; or we'll spiff it up for you with optional equipment.

The Peripheral Plate is a piece of bent metal with room for your keyboard, video monitor, two Philips decks or an audio cassette recorder. All at your fingertips. For organization.



Card rack swings out for service.

Naturally, our basics have down-to-earth prices, too. The Peripheral Plate is a mere $19.50 . . . the basic Basic Box goes for $45 (a little more depending on options).

Want to up your organization? It's simple. Just call or write the Digital Group for details.

Circle 45 on inquiry card.

the digital group

P.O. Box 6528 • Denver, Colorado 80206
(303) 777-7133

front panel logic was going to contain nearly 220 TTL integrated circuits if I implemented it with a conventional logic design. After that false start, I thought about a simplification made possible by a read only memory program, or "firmware" as it is sometimes called. I could replace most of the front panel logic with a program burned into a single read only memory integrated circuit. With a PROM program and 16 bytes of volatile programmable memory, the 6502 processor itself would operate the front panel of the system. The integrated circuit count for the front panel including the programmable read only memory and 16 bytes of volatile solid state memory was now reduced to just over 50 packages.

System Design Philosophy

During the process of designing the front panel, I worked out a total system design philosophy which goes like this:

- There would be a central processing unit and peripherals. The peripherals would be interfaced to the processor with a minimum of hardware by using memory address interfaces wherever possible.
- The system would be modular. A common backplane would be defined. Each module would be connected to the other modules through this backplane. Each socket on the backplane would be wired pin by pin to every other socket on the backplane.
- An address allocation map for the system would be defined. This would define addresses for hardware (peripherals), firmware (read only memory programs) and main programmable memory use.

The front panel design I had already created follows the first point of this philosophy quite well. It has several separate peripherals. Some are input devices, some are output devices, and some are a combination of both functions. Each is interfaced as a memory address and operated by firmware with a minimum of supporting hardware. Details of front panel operation will be discussed a little later in this article.

The physical arrangement of the design implements the details of the second point in the philosophy. The front panel assembly is connected to the top of a Vector prototyping card which contains the programmable read only memory with the front panel servicing routines. This card in turn plugs into the backplane bus which is implemented with a Vector card cage and edge connectors. By pulling the front panel card out of the backplane, the Kompuutar system can be isolated from the front panel completely. Similarly, the rest of the Kompuutar system is fabricated on Vector 3662 cards. Each card module contains one complete section of the system. These modules include the central processing unit card with the 6502 and bus interfacing chips, and a terminal interface card. Eventually 8 K byte programmable (volatile) memory cards will be part of the system. The cage I used has room for eight memory cards for a total of 64 K bytes. Since the backplane is wired from pin to corresponding pin of each socket, the cards can be placed in any available socket in the card cage. Table 1 shows the definitions of all the bus pins. In developing the system, I used an extender card plugged into the backplane so that I could have access to the various modules with an oscilloscope probe.

The third part of the design philosophy

Address Range	Type of Hardware	Usage of Region
0000 to 3FFF	Volatile programmable memory	This is the general programmable memory region for user applications programs. Locations 0000 to 01FF are dedicated to scratch pad and stack use by the architecture of the 6502 processor.
4000 to 6FFF	Unimplemented	This region is reserved for 12 K of general user memory expansion.
7000 to 73FF	TIM read only memory	When the TIM monitor interface card is in the system, this area is reserved.
7400 to 7FFF	Unimplemented	
8000	Scratch pad memory with external visibility	Current accumulator value maintained by front panel service program
8001	Scratch pad memory with external visibility	Current X register value
8002	Scratch pad memory with external visibility	Current Y register value
8003	Scratch pad memory with external visibility	Current processor flag values
8004	Scratch pad memory	
8005	Scratch pad memory with external visibility	Current stack pointer value
8006	Scratch pad program begins here	
8007 to 8008	Scratch pad	Current address register value, displayed through locations 8014 and 8015
8009 to 800C	Scratch pad program area	
800D	Scratch pad memory with external visibility	Current data at memory location in address register locations 8007 to 8008
800E-800F	Scratch pad	
8010	Read only data input	Front panel request register (see table 3)
8011	Read only data input	Low order address and data switch register
8012	Read only data input	High order address switch register
8013	Write only display	Flag data latch and binary display
8014	Write only display	Low order address display latch
8015	Write only display	High order address display latch
801F	Idle command address	References cause processor to idle
8020 to 80FF	Peripherals	Unimplemented hardware device addresses
8100 to EFFF	Unimplemented	
F000 to FDFF	Programmable read only memory allocations for systems programs	This area is expected to be used by interrupt service routines, utility subroutines and the like, programmed into read only memory parts.
FE00 to FFFF	Read only memory	This region is allocated to the firmware which controls the front panel. The 6502's interrupt vectors are programmed into the last portion (see listing 1).

Table 2: A memory allocation map for Kompuutar. When interfacing both peripherals and programming to a single memory address space, it helps to make a memory map to keep track of allocations.

Dynabyte builds the Great Memory

We cut up a Dynabyte 16k dynamic RAM board and constructed this pyramid to illustrate an important point: Dynabyte designs and builds memory boards with the same unmatched engineering ability and technical skill that went into Egypt's Great Pyramid.

One of the seven wonders of the ancient world, the Great Pyramid has been standing on the desert for an incredible 4,400 years. Although its enormous base covers 13 acres, it is perfectly square. Rising 450 feet, it is as tall as a 37 story building. Over 2.3 million blocks of stone were used, each averaging 2½ tons. Some weigh 16 tons. Despite their size, they fit together with a tolerance that is less than half the width of a human hair.

Dynabyte builds its 16k dynamic RAM boards with the same exceptional precision and care. Their reliability is as solid as a rock.

Dynabyte's design meets rigid industrial grade standards. The design is so good, in fact, that one of the largest, most experienced electronics manufacturers has tried to imitate it. (We were flattered but not surprised; we know how good it is.)

More than 1400 microcomputer owners also know how good it is. Dynabyte's 16k dynamic is running in more systems than any other dynamic memory on the market.

We select the best components we can buy to build the 16k dynamic, because solid parts make a solid memory. Our memory chips, for example, are factory prime from National Semiconductor.

Dynabyte was the first to deliver 16k dynamic RAM's assembled, tested and burned in. And at a price competitive with kits! Each board's complete function is confirmed by three stages of testing and a burn in cycle that runs 72 hours at 70° C (158°).

When we build them that solid we can guarantee them for a full year.

If a Dynabyte board ever needs repair, we provide factory service with a 24 hour turnaround for both warranty and non-warranty work.

The Dynabyte 16k dynamic has the widest compatibility of any dynamic memory. So it will work in your system.

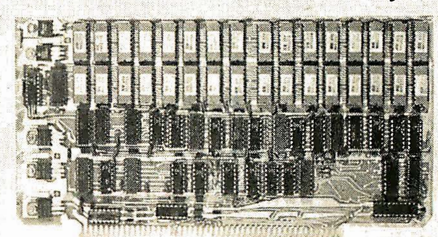
The Great Memory by Dynabyte is a solid buy. And an economical one. Effective October 1, the new Manufacturer's Suggested Price is reduced from $485 to $399.

Ask for the Great Memory by Dynabyte at your local computer store. I it isn't in stock, tell the owner that he missed another Dynabyte sale, and order direct. Telephone (415) 494-7817. Cable DYNABYTE. Or mail to Dynabyte, Inc., 4020 Fabian, Palo Alto, CA 94303.

Specifications: 16,384 bytes, National Semiconductor MM5271 chips, S-100 compatible, 350 nsec. access time, 550 nsec. cycle time, transparent refresh, no wait states for 2 MHz 8080 processor, on board clock, 5 watts power consumption, 1 MHz direct memory access, 16k addressing, solder masked, assembled with sockets, tested, burned in, guaranteed one year.

Dynabyte

Builders of the Great Memory



Circle 51 on inquiry card.

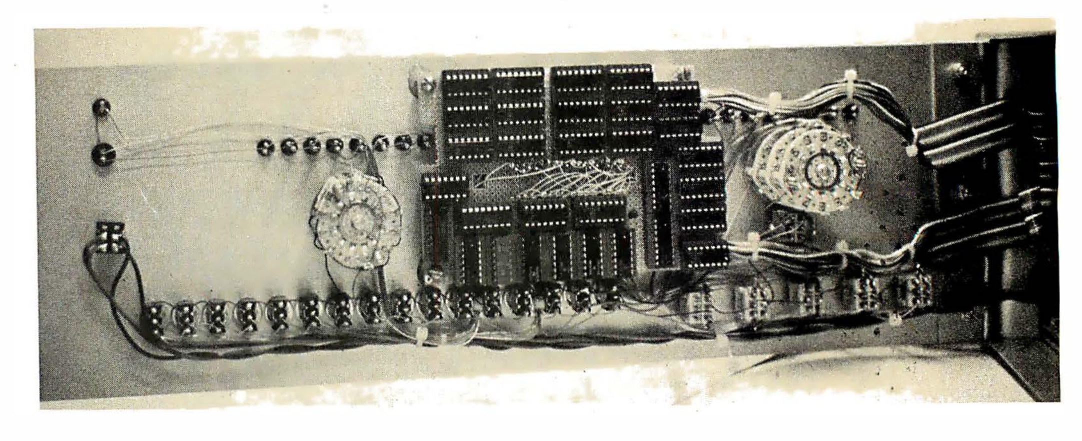


Photo 2: The reverse side of the front panel assembly for Kompuutar. The various switches, indicators and the front panel electronics board are seen in this picture. The P2-J2 cables run to the front panel interface module at right.

was implemented by picking address allocations. (See table 2 for a detailed list of the allocations.) I decided early in the project that 16 K bytes of memory would be a good start for general programming uses. I had to allocate this volatile user oriented programmable memory, as well as all the addresses for peripheral hardware and "firmware" read only memory programs. The address range of the 6502 is from 0 to 65,535 (0000 to FFFF in hexadecimal). Since the architecture of the chip itself uses addresses 0000 to 01FF for dedicated functions which must be in programmable memory, I assigned the 16 K byte block of main memory to the lowest part of the addressing range, from hexadecimal 0000 to 3FFF. I was interested in the possibility of occasionally using the MOS Technology TIM monitor, so I reserved locations 7000 to 73FF for use by that program's read only memory. I allocated the control panel scratch memory and peripheral ports starting at address 8000 hexadecimal, with the addresses starting at 8020 reserved for general peripheral use as I expand the system. At the end of the address range, I reserved the 4096 bytes from addresses F000 to FFFF for read only memory containing various systems routines. The high end of this range is reserved for the control panel support program and the interrupt vectors of the MOS Technology 6502 design.

Backplane

The backplane of the card cage (see table 1) carries the address bus, the bidirectional data bus, six vectored interrupt lines, and other functional signals as detailed in table 1. All signals that pass through the backplane are interpreted to be logical 1 or "true" in a low voltage (TTL 0) state. A high voltage (TTL 1) state is interpreted as a logical 0 or "false" state. Each module which connects to the backplane uses TTL inverting buffer circuits for signals sent or received. The +5 V (VCC) and ground (GND) connections are arranged on the card edge connectors such that by plugging a module into the backplane upside down, polarity to the card will not be reversed. This simple arrangement eliminates the need for keying the cards; while it prevents physical destruction of the card due to inadvertent reversal of orientation, the system should not, of course, be expected to work with one or more cards reversed relative to the balance of the cards in the system.

The vectored interrupt lines of the backplane are defined by some logic implemented on the central processing unit card (see figures 3). This card contains logic necessary to cause hardware vectoring of interrupt levels to one of the six possible interrupt service routines. The vectoring

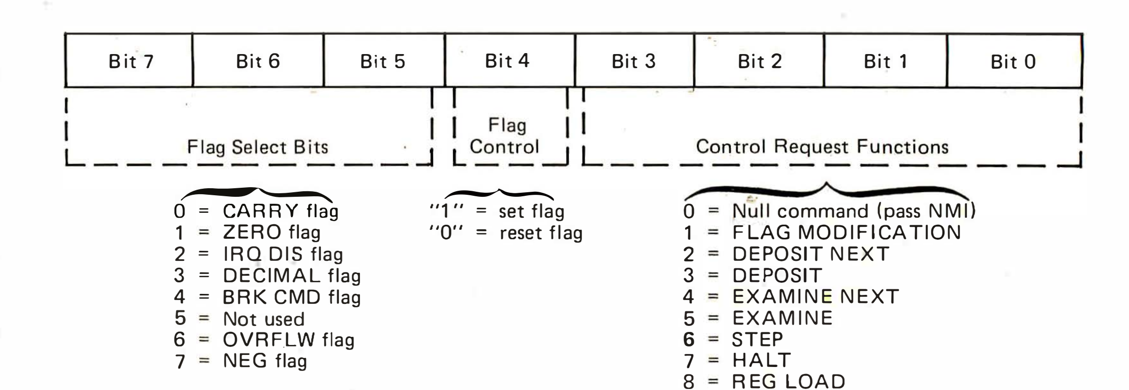


Table 3: Control request word layout. The control request word, located at address 8010 in memory address space, is an input to the processor with this format. It is used by the front panel service program of listing 1 to govern the operation of the panel based on settings of various switches.



The LATEST in Tape Systems





MODEL CC-8 $175.00

4800 BAUD CASSETTE RECORDER

An ASYNCHRONOUS NRZ type Recorder with remote motor start/stop. Error rate 10^8 at 4800 BAUD. Can be used from 110 to 4800 BAUD into a UART or "Bit Banger PIA" –no clocking required. This is not an audio recorder. It takes RS232 or TTL signals from the terminal or computer and gives back the same signals. No audio interface is used. Motor start/stop is manual or through TTL or RS232 signals.

Tape speed is 3.2"/second nominal; 1.6"/sec. optional. 110 volt, 60 Hz, 5 watts. (220 Volts on special order). Can use high quality audio cassettes (Philips Type) or certified data cassettes. Can be used in remote locations from a 12 Volt battery.

Recommended for DATA LOGGING, WORD PROCESSING, COMPUTER PROGRAM RELOADING and DATA STORAGE. Especially recommended for 6800 systems, 6502 systems, 1800 systems and beginners with the 8080 systems. Manual control except for motor start/stop. 6800 or 8080 software for file or record searching available on request with order. Used by major computer manufacturers, Bell Telephone and U.S. Government for program reloading and field servicing.

AVAILABILITY – Off the shelf.

6800 CONTROLLER for SWTP

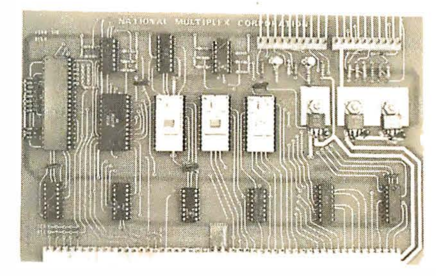
PROVIDES MONITOR AND TAPE SOFTWARE in EPROM. EXPANDS MIKBUG with 1 K of ADDITIONAL ROM PROGRAM.

This is a complete tape controller for the SWTP 6800 system. Has 3 K of EPROM space for your own programs. A 1 K ROM (2708) is provided with all tape and monitor functions. The ROM program is identical to our extensive 8080 ROM program.

Has one ACIA for one or two tape drives, one USART for an additional Serial port and a 4 bit parallel port for motor control. Will control one or two CC-8 or 3M3A drives with the software provided. Can be used with other tape drives controllable with 4 TTL bits if appropriate software changes are made.

Extra serial port is provided for your use with a second terminal or printer. (RS232, TTL or 20 ma)

The ROM program supplements the MIKBUG program and is entered automatically on reset.



AVAILABILITY–Off the Shelf.

$190.00, Tested & Assmb. ($160.00, Kit)

2SIO (R) CONTROLLER

$190.00 ($160.00 Kit)

PROVIDES MONITOR AND TAPE SOFTWARE in ROM TERMINAL and TAPE PORTS on SAME BOARD CONTROLS ONE or TWO TAPE UNITS (CC-8 or 3M3A)

This is a complete 8080, 8085, or Z80 system controller. It provides the terminal I/O (RS232, 20 mA, or TTL) and the data cartridge I/O, plus the motor controlling parallel I/O latches. Two kilobytes of on board ROM provide turn on and go control of your Altair or Imsai. NO MORE BOOTSTRAPPING. Loads and Dumps memory in hex on the terminal, formats tape cartridge files, has word processing and paper tape routines. Best of all, it has the search routines to locate files and records by means of six, five, and four letter strings. Just type in the file name and the recorder and software do the rest. Can be used in the BiSync (IBM), BiPhase (Phase encoded) or NRZ modes with suitable recorders and interfaces.

This is Revision 7 of this controller. This version features 2708 type EPROM's so that you can write your own software or relocate it as desired. One 2708 preprogrammed is supplied with the board. A socket is available for the second ROM allowing up to a full 2K of monitor programs.

Fits all S100 bus computers using 8080 or Z80 MPU's. Requires 2 MHz clock from bus. Cannot be used with audio cassettes without an interface. Cassette or cartridge inputs are RS232 level.

AVAILABILITY – Off the shelf.

Z 80 BOARD for SWTP COMPUTER: Now you can use the 8080/Z80 software programs in your SWTP 6800 machine. Replaces your MPU board with a Z80 and ROM so that you are up and running with your present SWTP memory and MPC card. $200 assembled and tested. ($160 kit)

AVAILABLE–November '77.

OVERSEAS: Export Version 220 volt 50 hz. Write factory or: Megatron-Datameg, 8011 Putzbrunn, Munchen, Germany; Nippon Automation 5-16-7 Shiba, Minato-Ku, Tokyo, Japan; Hobbydata, FACK 20012, Malmo, Sweden; G. Ashbee, 172 Ifield Road, London SW 10-9ag: Trintronics, Ltd., 186 Queen Street W., Toronto, Ontario, Canada; EBASA, Enrique Barges 17, Barcelona 14, Spain; ARIES, 7, rue Saint Phillipe du Roule, 75008 Paris; Microlem 20131, Milano, Italy; Eagle Electric, Capetown, S. Africa.

For U.P.S. delivery, add $3.00 Overseas and air shipments charges collect. N.J. Residents add 5% Sales Tax. WRITE or CALL for further information. Phone Orders on Master Charge and BankAmericard accepted.

National Multiplex Corporation

3474 Rand Avenue, South Plainfield NJ 07080 Box 288 Phone (201) 561-3600 TWX 710-997-9530

Circle 102 on inquiry card.

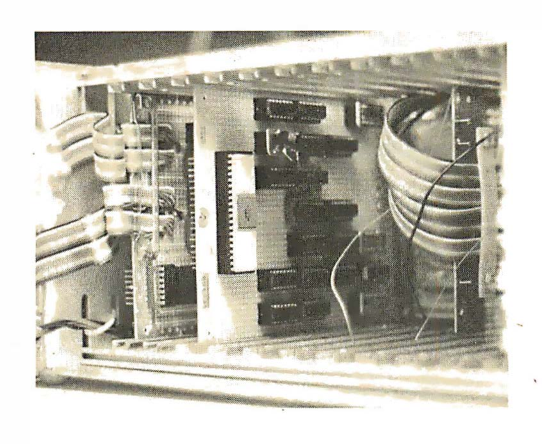


Photo 3: The front side of the backplane framework with card guides. This picture shows the front panel interface module at the left, the central processor module, a gap of several card slots, then the TIM interface module's edge with cables dangling.

is accomplished by using the selected service routine address for the interrupt vector requested by the 6502 processor during its interrupt sequence. The service routines are assumed to reside in programmable read only memory chips located in a separate module elsewhere on the backplane bus. The processor hardware also incorporates a priority arrangement of these six interrupts. Thus when two interrupts occur simultaneously the system has no problem: the higher priority one is serviced first. This becomes important when several interrupt driven peripherals are used with the system. An automatic reset function initializes the system when power is turned on, a separate interrupt for the 6502 which is supported on the processor board.

As an alternative to the front panel logic, the memory map of table 2 shows allocations for the MOS Technology TIM monitor integrated circuit, MCS6530-004. This "Terminal Interface Monitor" allows the user to use an ASCII serial device such as a Teletype or other terminal. In making a board to support TIM, I also included an 8 bit parallel interface to allow the possibility of using a high speed paper tape reader with Kompuutar. Details of the TIM module are shown in figures 4.

Front Panel Logic

Getting into more of the details of the system, I'll concentrate mainly on the place where I started my design, the front panel. The front panel logic is composed of input devices, output devices, 16 bytes of scratch pad memory, logic of the ready and nonmaskable interrupt timing, address decoders, switch debouncers, a data bus multiplexer, command encoder, line buffers and the control program in a programmable read only memory. The overall design of the front panel is found in figure 1, with details spread out in figures 1.1 thru 1.9. Photos 1 and 2 give further details.

There are four input sources of data in the front panel design. Each source is selected by the address decoding logic, which in turn allows the proper source to be input through the data bus multiplexer. The first source of input is the control request register. This source carries data from the command encoder, the flag selection switch, the flag modification switch and the register load switch. Table 3 shows the bit assignments of this source, which is located at hexadecimal address 8010 in memory address space.

The second source of input is the low

Text continued on page 112

Wire	Logic Diagram Mnemonic	Description	Wire	Logic Diagram Mnemonic	Description
1 to 4	+5 V	voltage source	21	X800X	address selection lines
5	D0	data bus lines	22	X800F	address selection lines
6	D1	data bus lines	23	X8013	address selection lines
7	D2	data bus lines	24	X8014	address selection lines
8	D3	data bus lines	25	SEL 1	multiplexer select lines
9	Φ1 CLK	phase 1 clock	26	SEL 2	multiplexer select lines
10	Φ2 CLK	phase 2 clock	27	BSOUT	multiplexer disable
11	IRQX	any interrupt pending	28	–	–
12	A3	address bus line	29	X8015	address selection lines
13	D4	data bus lines	30	XFEEA	address selection lines
14	D5	data bus lines	31	X801F	address selection lines
15	D6	data bus lines	32	–	–
16	D7	data bus lines	33	–	–
17	A2	address bus lines	34	REST	front panel reset
18	A1	address bus lines	35	RDY	ready
19	A0	address bus lines	36	NMI	nonmaskable interrupt
20	R/W	read/write	37 to 40	GND	ground

Table 4: Wiring list for the J2-P2 cable. This cable runs from the front panel interface board in the card cage to the front panel assembly, as seen in photos 2 and 3. [In the author's version of Kompuutar, the wiring was direct without use of a plug and jack; in order to simplify nomenclature in presenting the article, we've used a numerical indentification of signal paths as if a 40 wire cable and connectors had been used . . .CH]

Our prices are fantastic Our service...is better!

Specialization is the only way of doing a job well. S-100, Inc. is committed strictly to the needs of S-100 based minicomputer systems. We don't sell books, Teletypes, video terminals, or products that do not directly plug into an S-100 bus.

ABSOLUTELY NO BACK ORDERS! If we can't ship from stock, we will return your check. We deliver "off the shelf" — and we intend to inventory not only the best of the S-100 products, but also the items that are the best values as well. We don't plan to inventory everything made for the S-100 bus, but we intend to inventory anything you should have.

We stock mainframes but will concentrate on a variety of memory and interface boards, offering the most popular and the obscure.

If it's made for the S-100 bus, call or write us first... second... or last — but be sure to get our prices before you buy! Write for our complete catalog.

— TYPICAL PRICES ON SOME OF OUR ITEMS —

Item	List	OUR PRICE
IMSAI 22-SLOT MOTHERBOARD, KIT	(List $ 751.00)	$ 589.95
Z-80 CPU BOARDS—		
Our Kit		144.95
Digital Innovations (Similar to TDL)	(List $ 169.95)	149.95
TDL ZPU	(List $ 269.95)	239.95
S-100 SYSTEM, complete, all self-contained and partially assembled, includes keyboard, power supply, 10-slot motherboard, Z-80 CPU, 16K of RAM, I/O board with audio cassette, monitor in PROM, and 8K BASIC—all for ONLY		1,495.00
4K S-100 MEMORY BOARD, low-power, 450ns		89.95
8K S-100 MEMORY BOARD, low-power 450ns		144.95
16K S-100 STATIC MEMORY BOARD		439.95
16K S-100 DYNAMIC MEMORY BOARD (assembled and tested)		299.95
CYBERCOM VIDEO BOARD VB1A	(List $ 189.00)	149.95
NORTH STAR MICRO DISK, Kit	(List $ 699.00)	594.95

All Processor Technology and Cromenco items and TDL Software — **LESS 10%**

We distribute S-100 items from all major manufacturers and from many smaller ones. Unless otherwise specified, all boards or kits are fully socketed (sockets are extra for IMSAI boards or kits).

An organization dedicated to the users of the S-100 bus **S-100, Inc**

Circle 118 on inquiry card.

7 WHITE PLACE
CLARK, NEW JERSEY 07066
(201) 382-1318

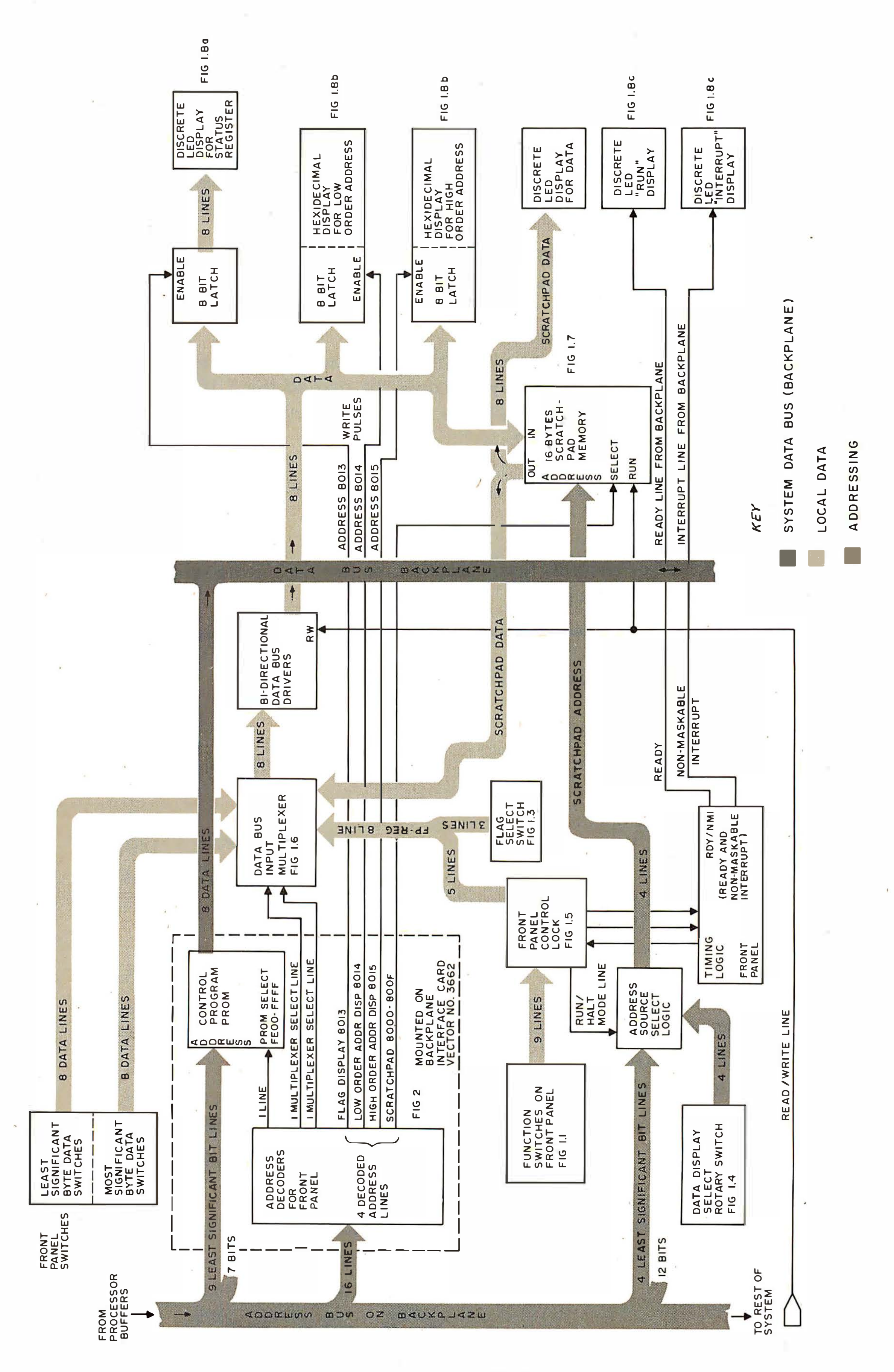


Figure 1: Block diagram of Kompuutar's front panel logic. The Kompuutar design uses a read only memory program to manipulate the contents of memory interactively using function switch inputs and solid state display outputs. This diagram serves as a functional road map to the various components of the display and its interface board.

16384 BYTES for $485.00

assembled (with sockets) : tested - burned-in - guaranteed

A new high in S100 bus memory cost effectiveness. Fully assembled (with sockets), tested, burned-in and guaranteed. 4Kx1 dynamic memory chips (the same ones used by the ton in IBM compatible memory systems) combined with self contained control logic, yield a memory system with these features:

- Low power consumption, total board 5 watts.
- Transparent refresh, which means the memory looks static to the outside world.
- No waiting. In fact, XRDY is not even connected to the memory.
- Full DMA capability.
- Reliable, low level clock and control signals.
- Three full days testing at 70° C (185° F).

SPECIFICATIONS

Capacity:	16384 bytes
Addressing:	16K boundaries
Bus structure:	S100 - Plug compatible with IMSAI 8080, POLY 88, ALTAIR 8800, BYTE-8, SOL
Address and Data Buffering:	<200 uA, special high impedance buffers - less than one low power Schottky load
Access time:	350 nSec
Memory chips:	MM 5271 (National Semiconductor and others) 4K dynamic

Dynabyte Inc.

4020 Fabian Way, Palo Alto Ca. 94303

for more information call or write to:

R.H.S. MARKETING
2233 El Camino Real
Palo Alto, California 94306
(415) 321-6639

DEALER INQUIRES INVITED
B of A & Mastercharge accepted

Dynabyte® brings to the S100 Bus a state of the art, industrial quality memory system. 16K on a single board for $485.00, Guaranteed for 1 year.

Circle 122 on inquiry card.

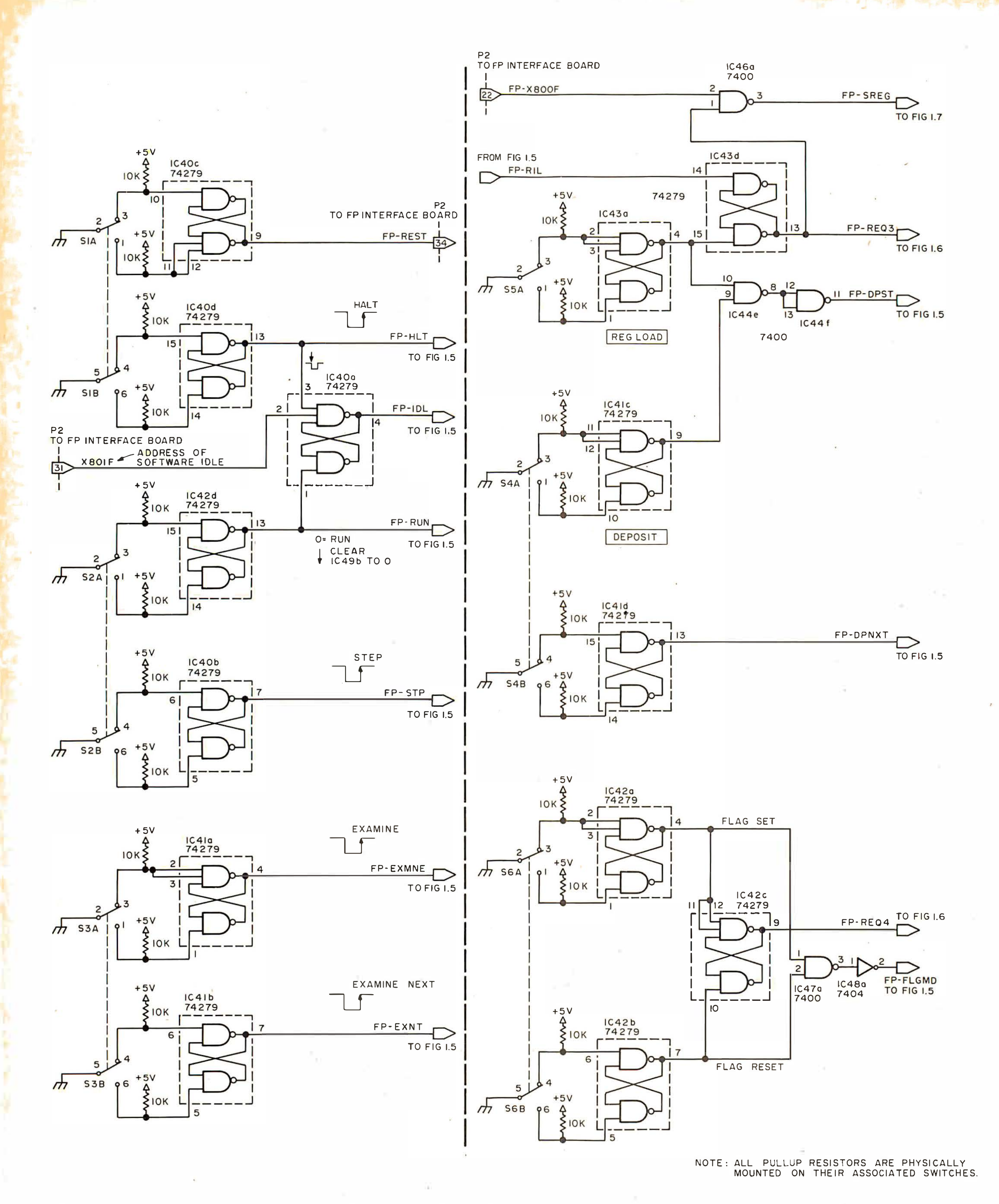


Figure 1.1: Switch debouncing logic. This is a detail logic diagram suitable for construction of Kompuutar. As in all the logic of this design, all resistors are 1/4 W unless otherwise noted, and standard TTL integrated circuits are used for miscellaneous functions. Debouncing is done with set-reset flip flops contained in the 74279 part, which we have noted in the discrete logic form internal to dotted lines. The flip flops can be wired out of gates (7400, 7410) if desired, should the 74279 be unavailable in the builder's parts bin. Integrated circuit power wiring for the entire design is summarized by IC number in table 5.

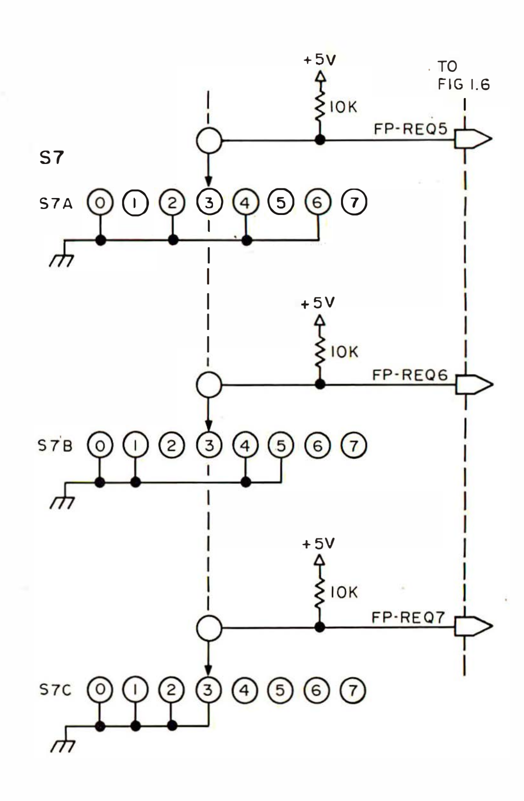


Figure 1.2: Flag selection switch. The control panel service program of Kompuutar uses the binary encoded 3 bit value on the output of this switch to determine which processor flag is to be set or reset using an appropriate function selection. This switch is a rotary switch which has three poles and eight positions.

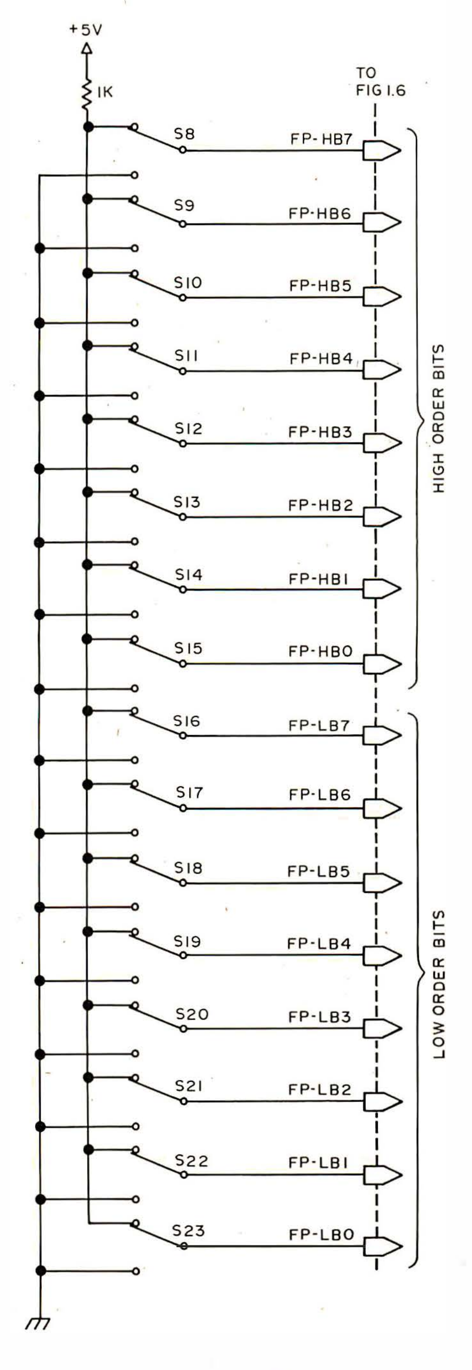
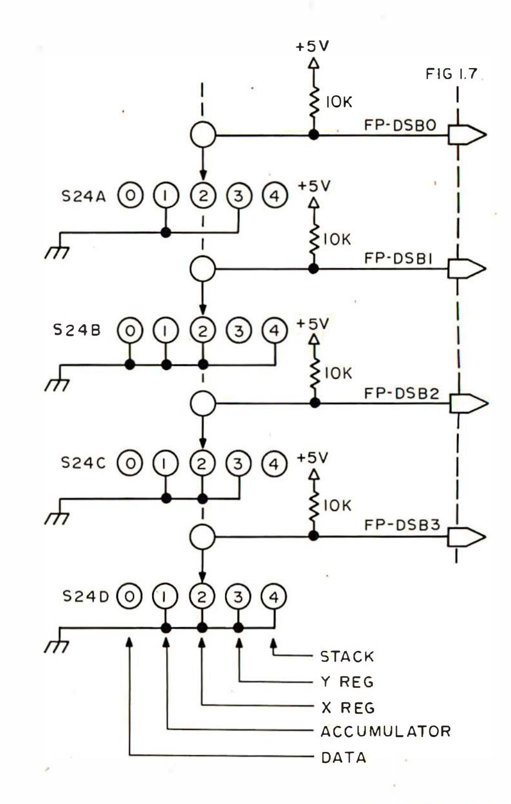


Figure 1.3: Front panel data entry switches. The entry of static data is accomplished by 16 toggle switches. These switches are connected to either logical 1 (+5 V through a 1 K resistor) or logical 0 (ground).

Figure 1.4: Display data selection switch. The control panel service program uses the binary encoded 4 bit value on the output of five possible words for default display from the control panel scratch pad located at addresses 8000 to 800F. The five addresses selected are for the accumulator (0), X index (1), Y index (2), stack register (5) or data register (D).



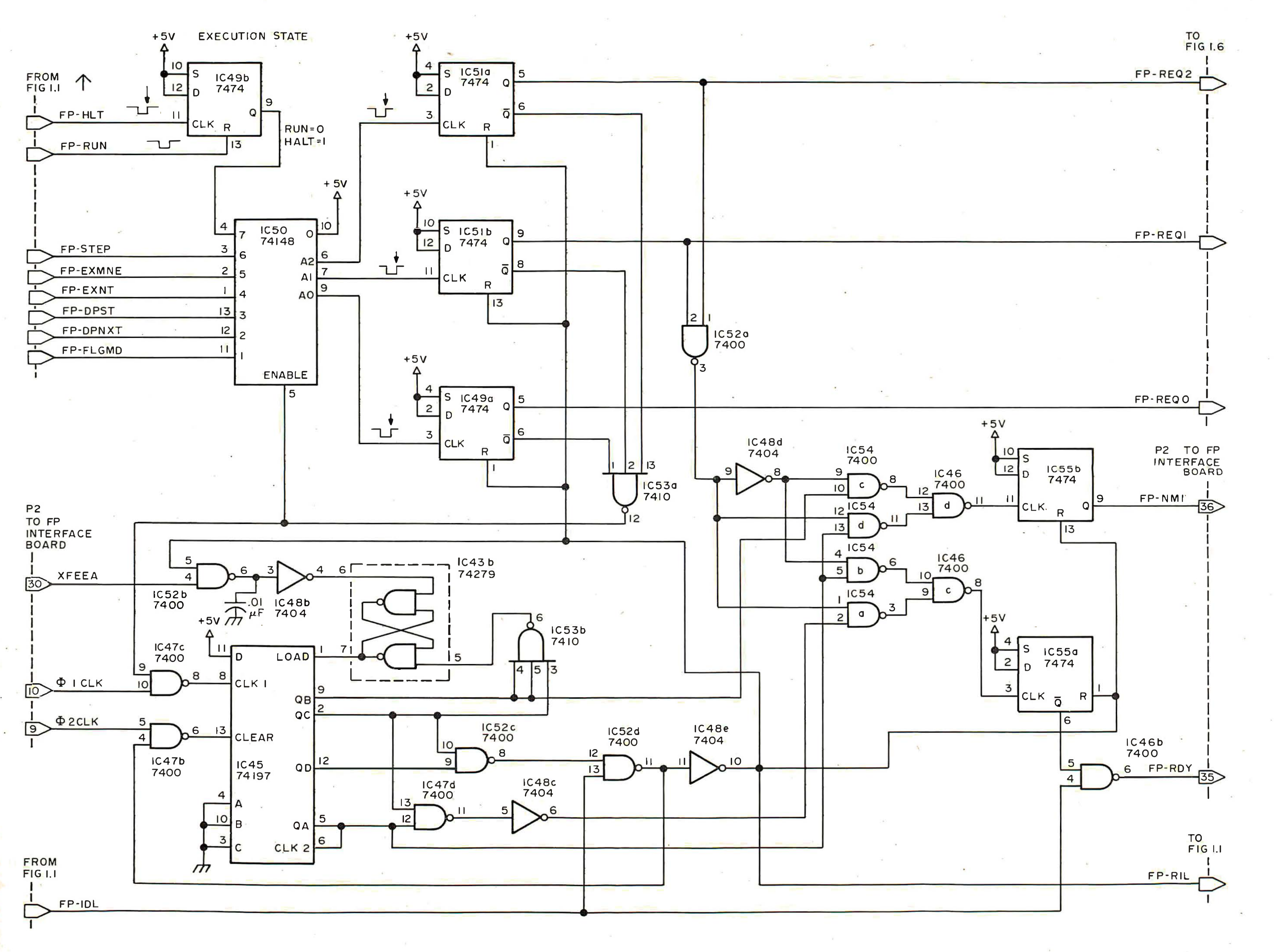

EXECUTION STATE
FROM FIG 1.1
FP-HLT
FP-RUN
IC49b 7474
RUN=0
HALT=1
IC50 74148
ENABLE
FP-STEP
FP-EXMNE
FP-EXNT
FP-DPST
FP-DPNXT
FP-FLGMD
IC51a 7474
IC51b 7474
IC49a 7474
IC52a 7400
IC53a 7410
TO FIG 1.6
FP-REQ2
FP-REQ1
FP-REQ0
IC48d 7404
IC54 7400
IC46 7400
IC55b 7474
P2 TO FP INTERFACE BOARD
FP-NMI
36
IC55a 7474
IC46b 7400
FP-RDY
35
TO FIG 1.1
FP-RIL
P2 TO FP INTERFACE BOARD
XFEEA
30
IC52b 7400
.01 µF
IC48b 7404
IC43b 74279
IC53b 7410
IC47c 7400
Φ 1 CLK
10
Φ 2CLK
9
IC47b 7400
IC45 74197
LOAD
CLK 1
CLEAR
CLK 2
IC52c 7400
IC52d 7400
IC48e 7404
IC47d 7400
IC48c 7404
+5V
FROM FIG 1.1
FP-IDL

Figure 1.5: Control logic for front panel functions. This logic generates the function request code (read from address 8010 bits 0 to 2), and controls the NMI line of the 6502 to implement single step execution of the processor.

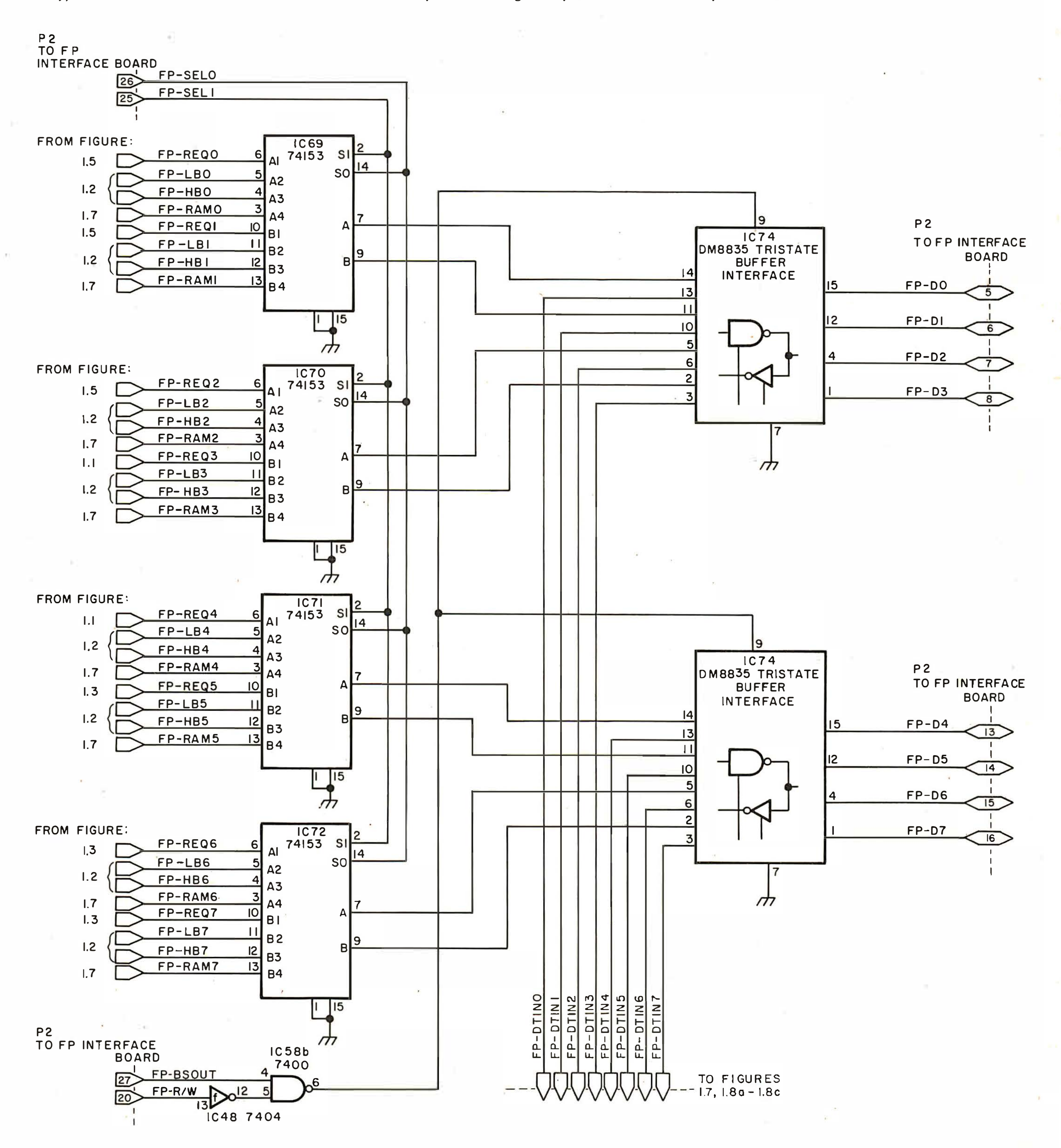


Figure 1.6: Data source multiplexer and bus interface. The sources of data read from the front panel logic are four: the two 8 bit data entry switch registers of figure 1.3, the 8 bit control request word from figure 1.5, and the output of the scratch pad programmable memory (lines labeled "RAM") from figure 1.7. These are selected by a 2 bit addressing code generated on the front panel interface board of figures 2.

Text continued from page 102

order byte of the data entry switches. The eight toggle switches of this switch register are used to enter a byte into the data bus or into the least significant byte of the address register which is maintained by the control panel program in its scratch pad. These toggle switches are located at address 8011 in memory address space.

The third source of data for the control panel program is the set of toggle switches which define the most significant byte of an address. These eight switches are located at address 8012, and are only used for address inputs.

The last source of data is the output of the 16 byte scratch pad memory in the control panel. The scratch pad responds to addresses 8000 thru 800F.

The address decoding logic is found in figure 2.1. The outputs of this decoding logic include miscellaneous individual address selections, plus the selection signals which are used to control the data input multiplexer found in figure 1.6. The selection signals are generated by the priority encoder IC35, and are used to pick one of the four sources for routing to the bus interface gates IC73 and IC74. These gates connect to the backplane data bus from the front panel via P2's connecting cable between the front panel and the front panel interface board.

The front panel also includes several possible outputs for data. In addition to the input possible from the scratch pad, the processor can address and write data to the scratch pad in any one of the locations 8000 to 800F. The actual contents of the data in the scratch pad can be displayed for addresses 8000, 8001, 8002, 8003, and 800D by moving the rotary switch S24. This switch (see figure 1.4)

Text continued on page 116

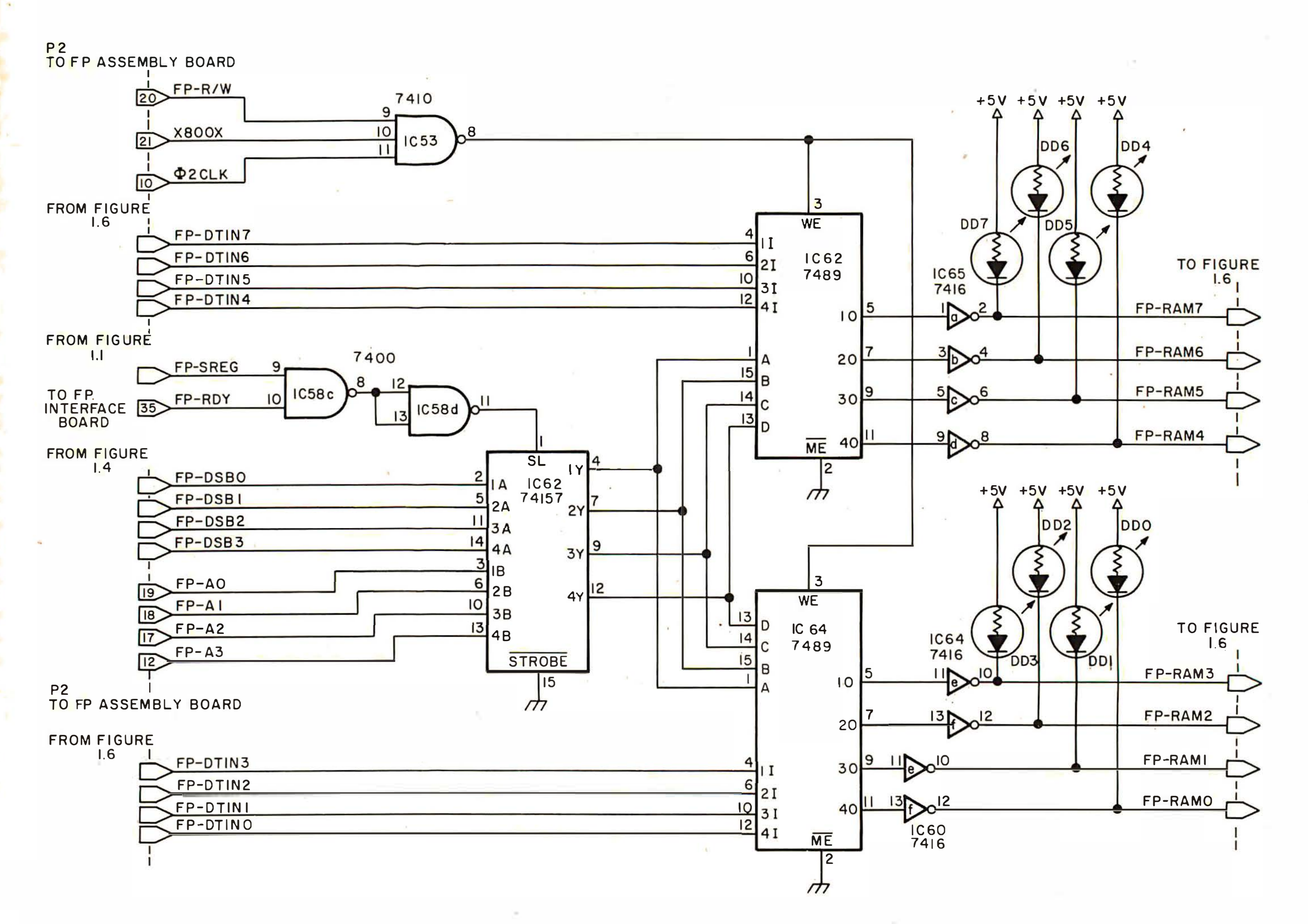
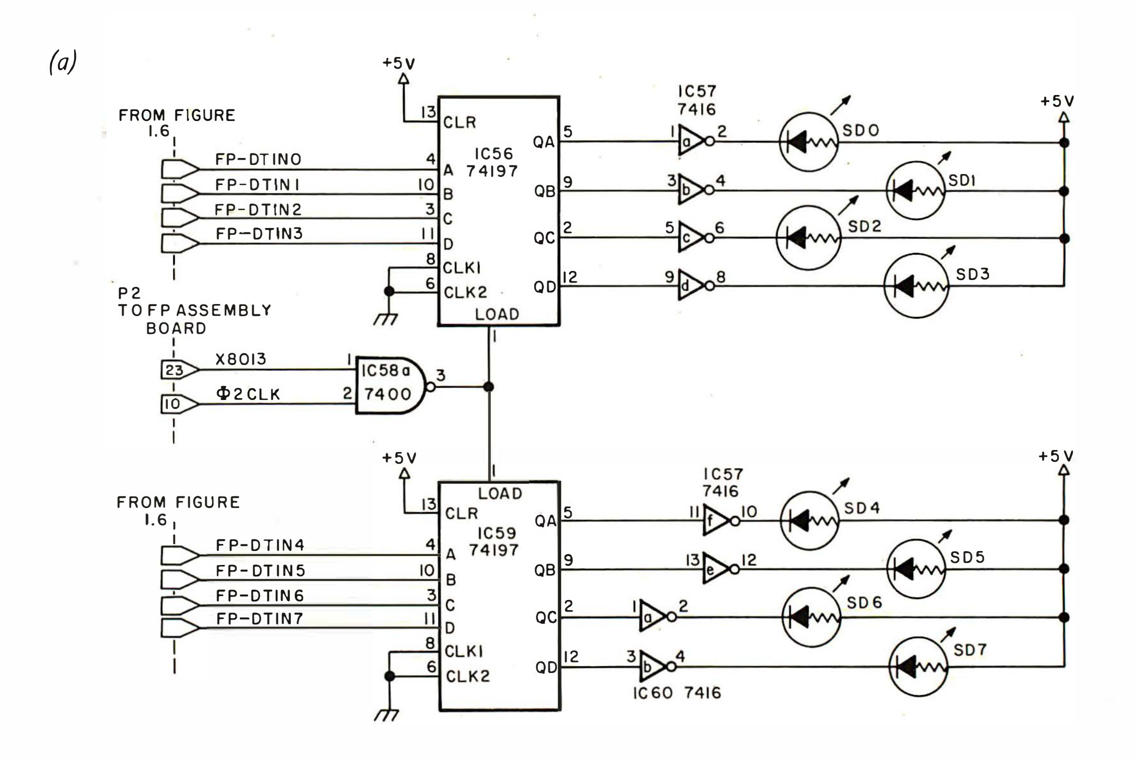


Figure 1.7: Front panel scratch pad programmable memory. The front panel implements a 16 byte scratch pad programmable memory at addresses 8000 to 800F. This memory is used for data storage and for storage of a scratch pad program segment which is modified during execution of the front panel service routines.

Circle 88 on inquiry card.

(a)



(b)

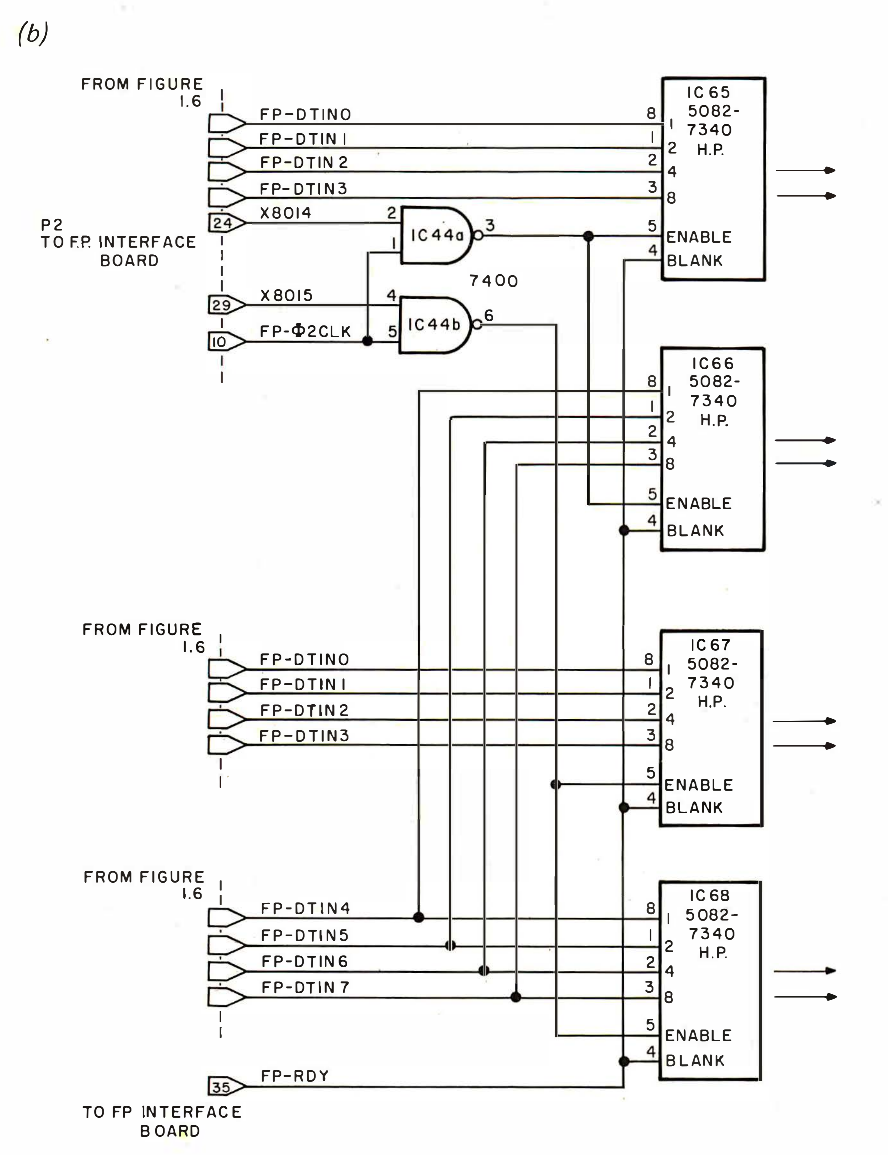
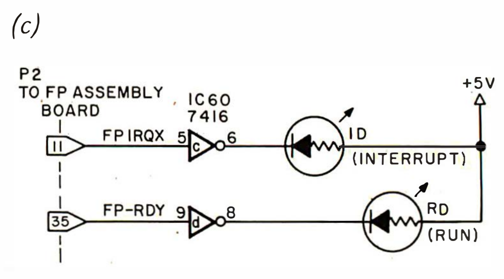
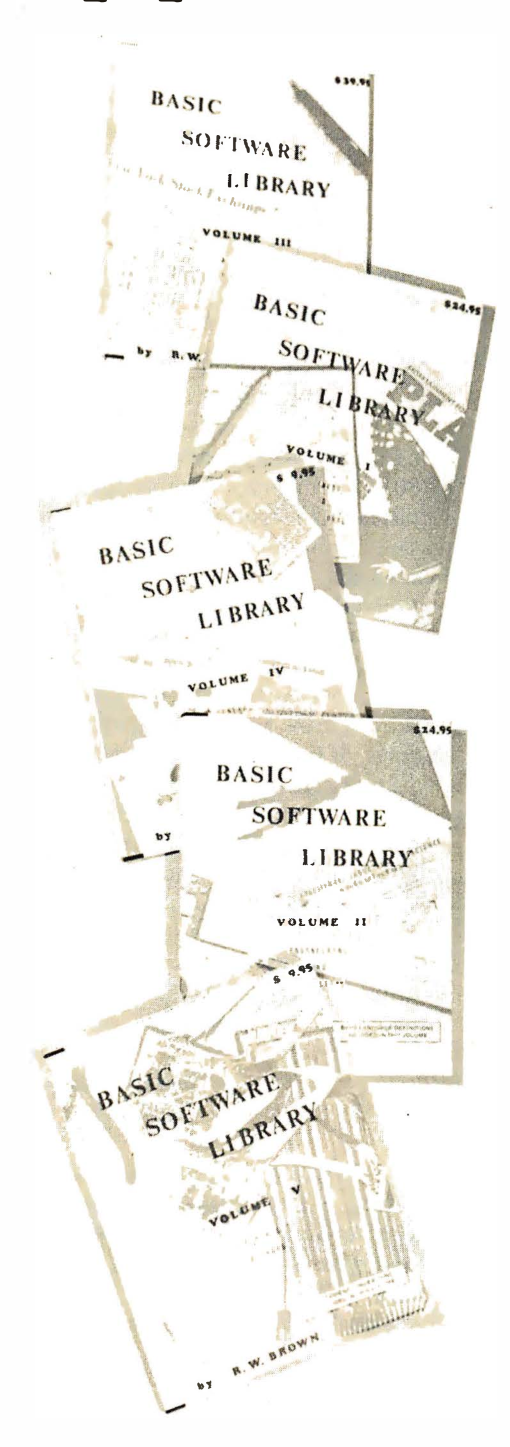


Figure 1.8: Displays. The front panel displays are detailed here: (a) is the output latch used to drive LED indicators for the eight flag bits, located at hexadecimal address 8013. At (b) are the four digits of hexadecimal address lamp display, addressed at locations 8014 and 8015. These displays incorporate latching logic as well as the needed decoding of 4 bit hexadecimal patterns into an array of LED dots. And at (c) are two miscellaneous indicators for the front panel.

(c)



Application Software!



You can buy software from anybody - but ours works in your system.

We only sell one product, Quality.

We have been in business for over nine years building a reputation for providing a quality product at nominal prices — NOT what the traffic will bear. Our software is:

- **Versatile** — as most programs allow for multiple modes of operation.
- **Tutorial** — as each program is self prompting and leads you through the program (most have very detailed instructions contained right in their source code).
- **Comprehensive** — as an example our PSD program not only computes Power Spectral Densities but also includes FFT's, Inverse-transforms, Windowing, Sliding Windows, simultaneous FFT's variable data sizes, etc. and as a last word our software is:
- **Readable** — as all of our programs are reproduced full size for ease in reading.
- **Virtually Machine Independent** — these programs are written in a subset of Dartmouth Basic but are not oriented for any one particular system. Just in case your Basic might not use one of our functions we have included an appendix in Volume V which gives conversion algorithms for 19 different Basic's; thats right, just look it up and make the substitution for your particular version. If you would like to convert your favorite program in to Fortran or APL or any other language, the appendix in Volume II will define the statements and their parameters as used in our programs.

Over 85% of our programs in the first five volumes will execute in most 8K Basic's with 16K of free user RAM. If you only have 4K Basic, because of its' lack of string functions only about 60% of our programs in Volumes I thru V would be useable, however they should execute in only 8K of user RAM.

All of our programs are available on machine readable media. For those that have specific needs, we can tailor any of our programs for you or we can write one to fit your specific needs.

Vol. I — $24.95 Bookkeeping Games Pictures	**Vol. III — $39.95** Advanced Business Billing, Inventory Investments Payroll
Vol. II — $24.95 Math/Engineering Plotting/Statistics Basic Statement Def.	**Vol. IV — $9.95** General Purpose **Vol. V — $9.95** Experimenter's Program



SCIENTIFIC RESEARCH

1712-B Farmington Court, Crofton, MD 21114



Phone orders call (800) 638-9194
Information and Maryland residents call (301)721-1148

master charge

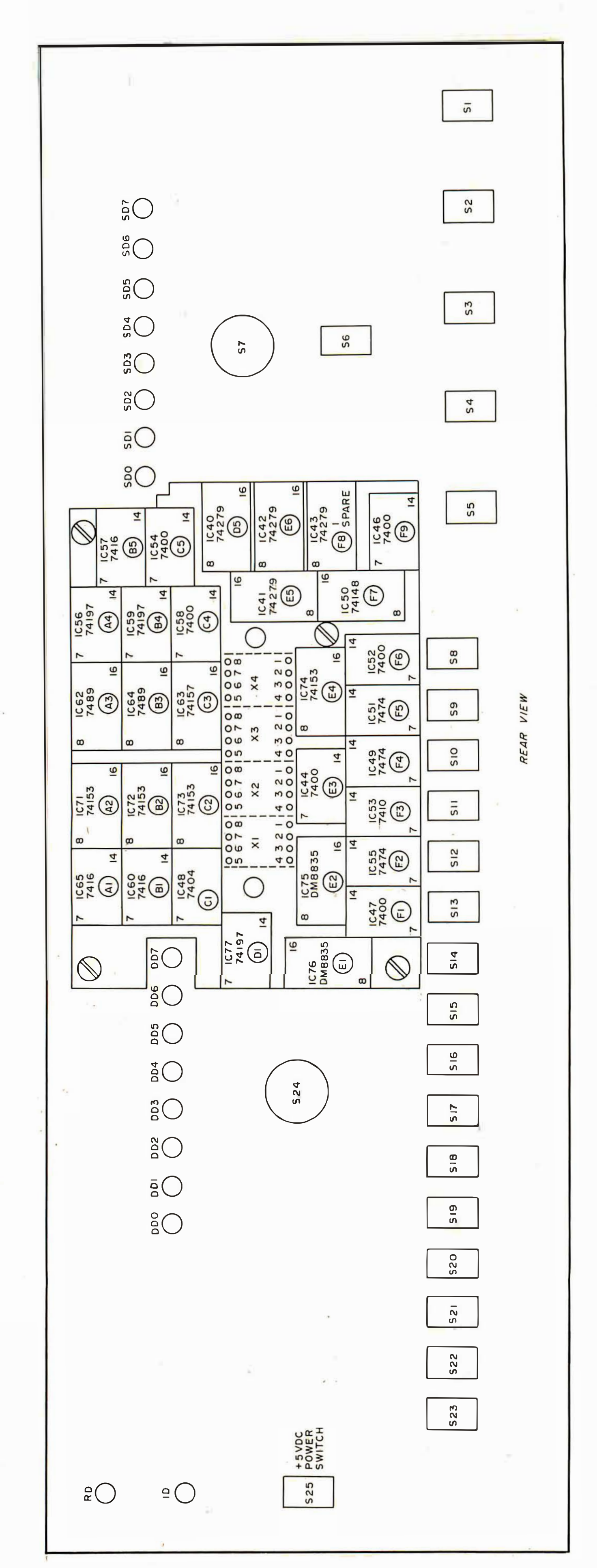
BANKAMERICARD

Add $1.50 per volume handling, all domestic shipments sent U.P.S. except APO and P.O. Box which go parcel post. Foreign orders add $8.00/volume for air shipment and make payable in U.S. dollars only.

AVAILABLE AT MOST COMPUTER STORES

Master Charge and Bank Americard accepted.

Our Software is copyrighted and may not be reproduced or sold.



Text continued from page 112

defines an address value which is presented to the 7489 scratch pad memories IC61 and IC63 by 74157 multipelexer IC62 when the scratch pads are not being referenced by the processor. Since the control panel program references the scratch pad only occasionally, the normal state is an address selected by S24 determining which of the five scratch pad locations is seen in the display lamps for scratch pad data. Thus the scratch pad memory has several potentially visible bytes of memory address space and acts as an output device.

A second output device is an 8 bit data latch whose outputs activate eight discrete LED devices. This device, located at address 8013 in memory address space, is used to display the processor's status register bits. The front panel control program is responsible for maintaining current information in this display (as is the case with all the display outputs).

Address display information is latched into four digits of hexadecimal display provided by Hewlett-Packard HP5082-7340 parts, IC65, IC66, IC67 and IC68 in figure 1.8b. Each of these displays has a built-in 4 bit data latch which retains information defined by writing to the memory locations 8014 (low order) and 8015 (high order).

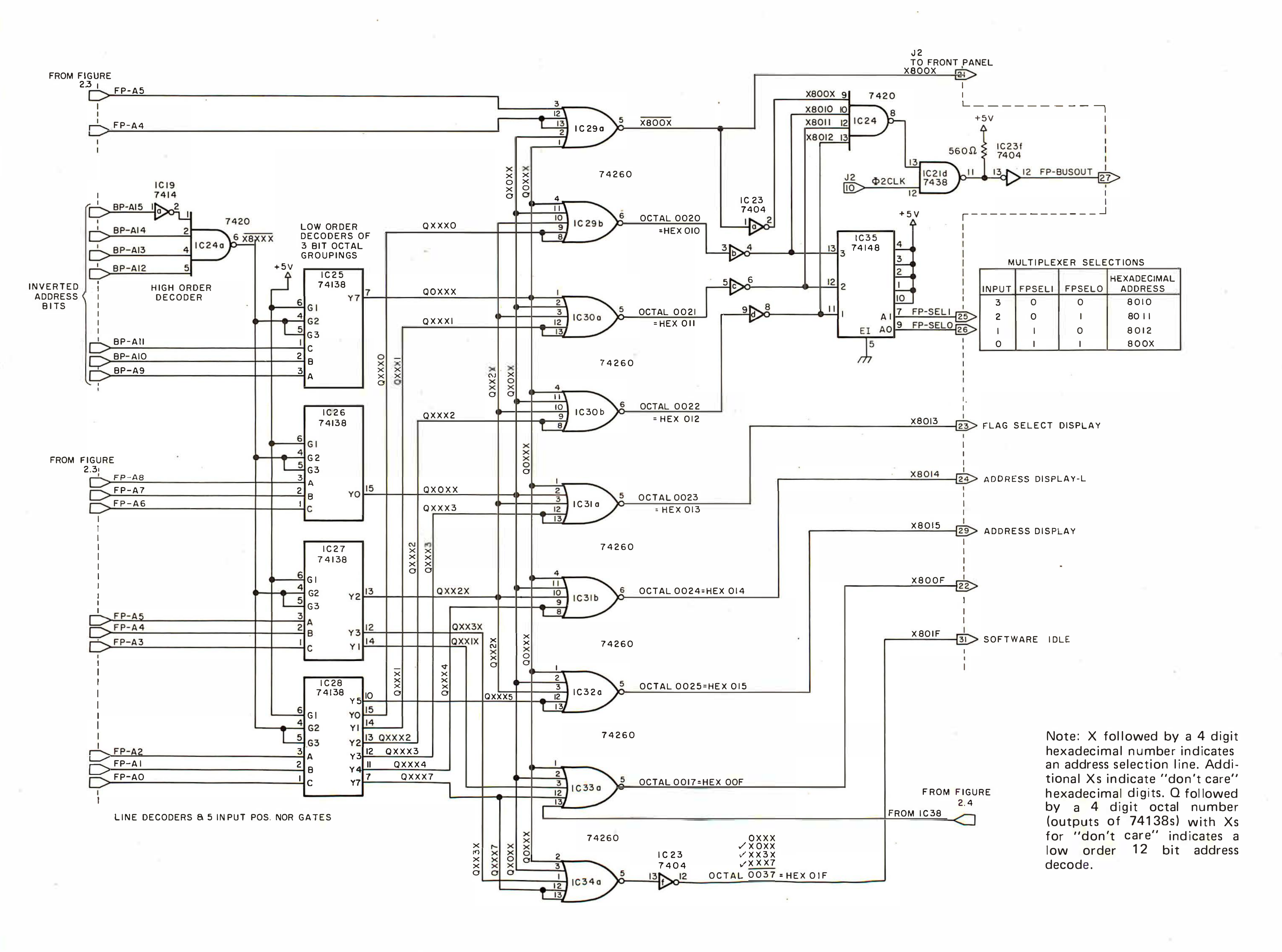
Two miscellaneous indicators are also included in the design. These single bit LED displays are connected to the processor's ready (RDY) and main interrupt (IRQ) lines. The RUN indicator lights

Text continued on page 119

Figure 1.9: Front panel mechanical layout. This is a detail drawing to scale of the physical layout of the front panel as seen in the photographs, along with the front panel electronics board which is mounted on standoffs behind the panel.

NOTES:

1. All ICs except locations E1 and E2 are Texas Instruments SN74XXX series TTL logic. E1 and E2 are three-state bidirectional bus drivers made by National.
2. Switches S1 thru S6 are Alco model MTF-206SA.
3. Switches S8 thru S23 are Alco model MTF-106D.
4. Switch S7 is a Centralab model PA-2009 rotary.
5. Switch S24 is a Centralab model PA-2011 rotary.
6. RD, ID, DD0 to DD7 and SD0 to SD7 are discrete LED displays. HP model 5082-4860.
7. X1-X4 are dot matrix hexadecimal LED displays. HP model 5082-7340.
8. All pull up resistors shown on schematics in conjunction with front panel switches are mounted on those switches.


Note: X followed by a 4 digit hexadecimal number indicates an address selection line. Additional Xs indicate "don't care" hexadecimal digits. Q followed by a 4 digit octal number (outputs of 74138s) with Xs for "don't care" indicates a low order 12 bit address decode.
MULTIPLEXER SELECTIONS
INPUT FPSEL1 FPSEL0 HEXADECIMAL ADDRESS
3 0 0 8010
2 0 1 8011
1 1 0 8012
0 1 1 800X
J2 TO FRONT PANEL
FP-BUSOUT
FLAG SELECT DISPLAY
ADDRESS DISPLAY-L
ADDRESS DISPLAY
SOFTWARE IDLE
FROM FIGURE 2.4
FROM IC38
FROM FIGURE 2.3
INVERTED ADDRESS BITS
HIGH ORDER DECODER
LOW ORDER DECODERS OF 3 BIT OCTAL GROUPINGS
LINE DECODERS & 5 INPUT POS. NOR GATES
OCTAL 0020 =HEX 010
OCTAL 0021 =HEX 011
OCTAL 0022 =HEX 012
OCTAL 0023 =HEX 013
OCTAL 0024=HEX 014
OCTAL 0025=HEX 015
OCTAL 0017=HEX 00F
OCTAL 0037 =HEX 01F

Text continued from page 116

up when the machine is in the run mode, and the interrupt indicator remains lit while an interrupt is pending. Interrupt control logic is contained in the devices requesting an interrupt, with the processor's priority encoder defining the backplane signal IRQX (backplane pin 12) which indicates that some interrupt is pending (and also signals the processor through its IRQ input, pin 4.) Thus when interrupt driven IO is used, the interrupt indicator lamp will flicker if appreciable interrupt processing wait states occur as various devices request attention.

Other Front Panel Functions

The front panel logic includes logic of the ready (RDY) and nonmaskable interrupt (NMI) timing, shown in detail in figure 1.5. These lines are used to generate signals which affect the processor in a manner very similar to interrupts. The HALT and single STEP switch are used to generate signals for these lines. This timing logic causes the 6502 processor and its control program to implement fairly conventional single stepping and program halt or restart functions. HALT or STEP switches are used to cause the processor to complete the present instruction, then execute one more instruction. Any other switch activated on the front panel causes this logic to allow the processor to complete only its present instruction. The hardware protocol of this logic locks out all front panel functions when the processor is running, except for the HALT switch.

The front panel's interfaces to human fingers are through various function switches. These switches are debounced using set-reset flip flops which come four to a package in the 74279 part. The debouncing logic guarantees that only one pulse is received for each activation of a switch.

Getting Kompuutar Into Operation

The operation of the front panel's control logic with respect to the actual processor

Figure 2.2: Buffering of processor control signals at the front panel interface module.

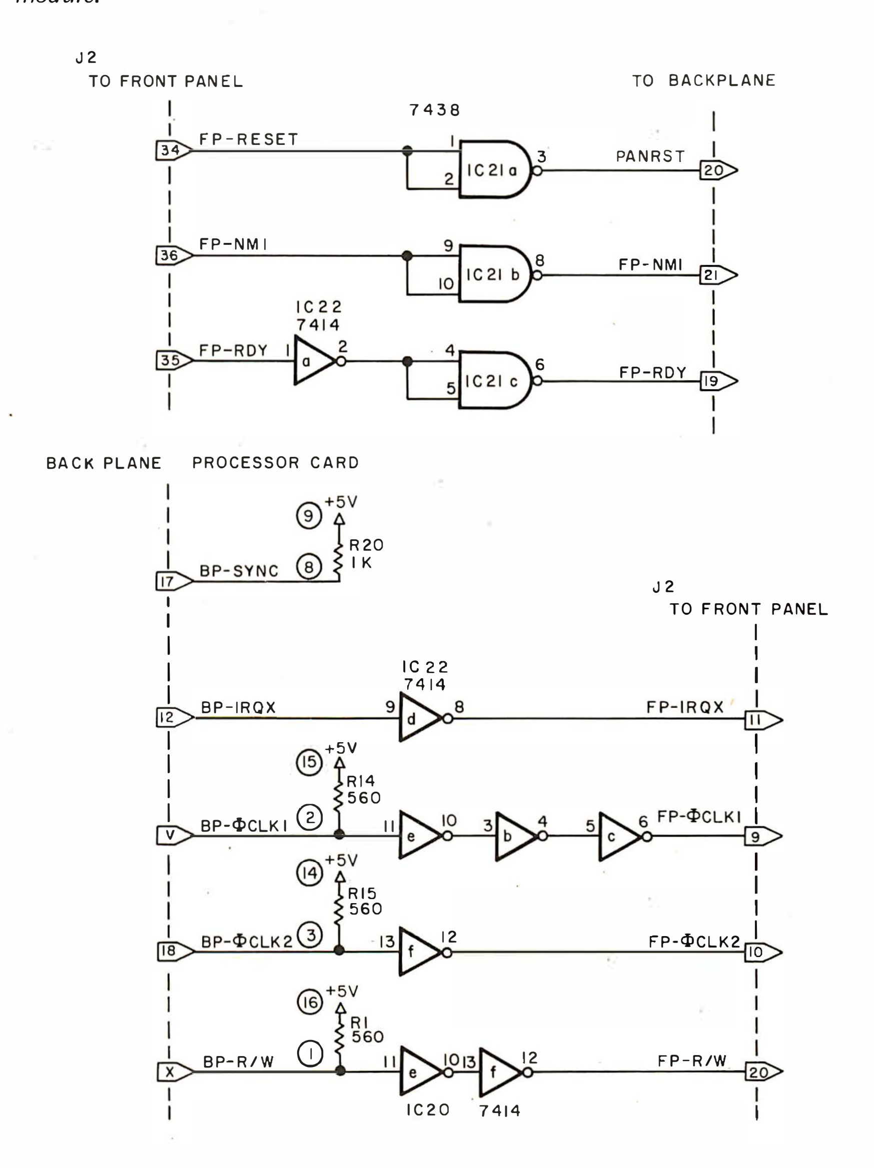


Figure 2.1: Front panel interface module address decode logic. This logic decodes the several addresses in the 8000 to 801F range which are used by the front panel design of Kompuutar. Since 3 bit decoders are used, octal intermediate terms are used to symbolize the outputs of the 74138s prior to logical sums performed by the 74260 OR gates. Outputs of the circuit are discrete select lines for several addresses, plus two source selection lines for the data bus input multiplexer of figure 1.6.

You Asked For It!

You asked for the Intel 8080A CPU . . . . We gave you the VECTOR 1

You asked for a real-time clock . . . . We gave you the VECTOR 1

You asked for eight level
vectored priority interrupts . . . . We gave you the VECTOR 1

You asked for a jump-on-reset
and resident monitor . . . . We gave you the VECTOR 1

You asked for the S-100 bus . . . . We gave you the VECTOR 1

You asked for an 18 slot
fully shielded motherboard . . . . We gave you the VECTOR 1

You asked for a
rugged, reliable chassis . . . . We gave you the VECTOR 1

Then . . . You asked for the compactness and convenience of installing your mini-floppy disk directly in the front panel . . . . NOW . . . We give you the **VECTOR 1+**



Not affiliated with Vector General, Inc.

790 Hampshire Rd., A + B, Westlake Village, CA 91361 Telephone: (805) 497-0733

can be illustrated by walking verbally through a typical sequence of operations. First, let's assume that the machine is in RUN mode, which is indicated by a low level on the output of the execution state flip flop, IC49b pin 9. This is the normal situation for a fully executing 6502 program contained in the system's main programmable memory region. Next, press the front panel's HALT switch, S1. Upon release of the HALT switch the debounce logic completes one HALT pulse which is processed by the command encoding logic of figure 1.5. When the HALT line makes

Text continued on page 128

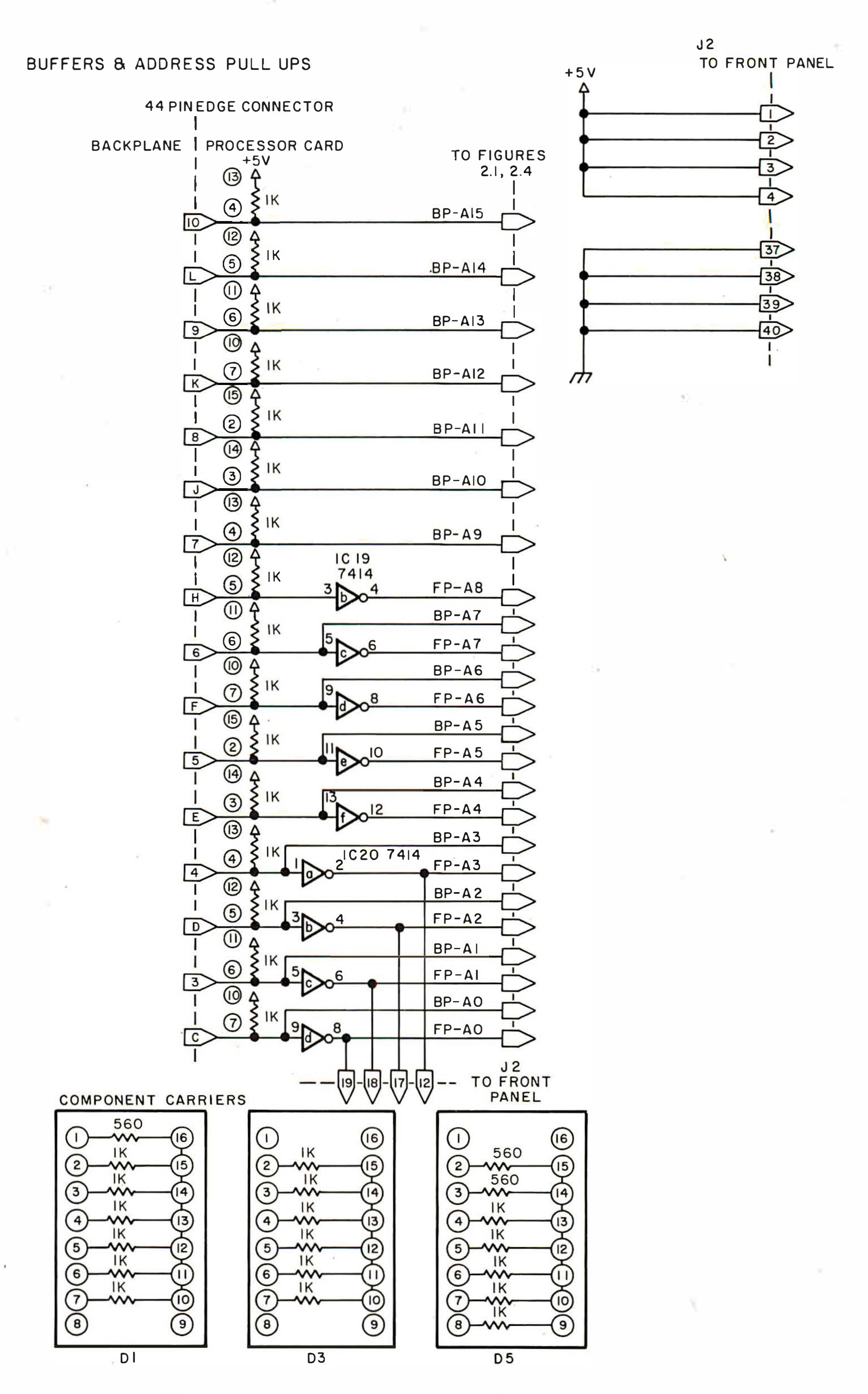
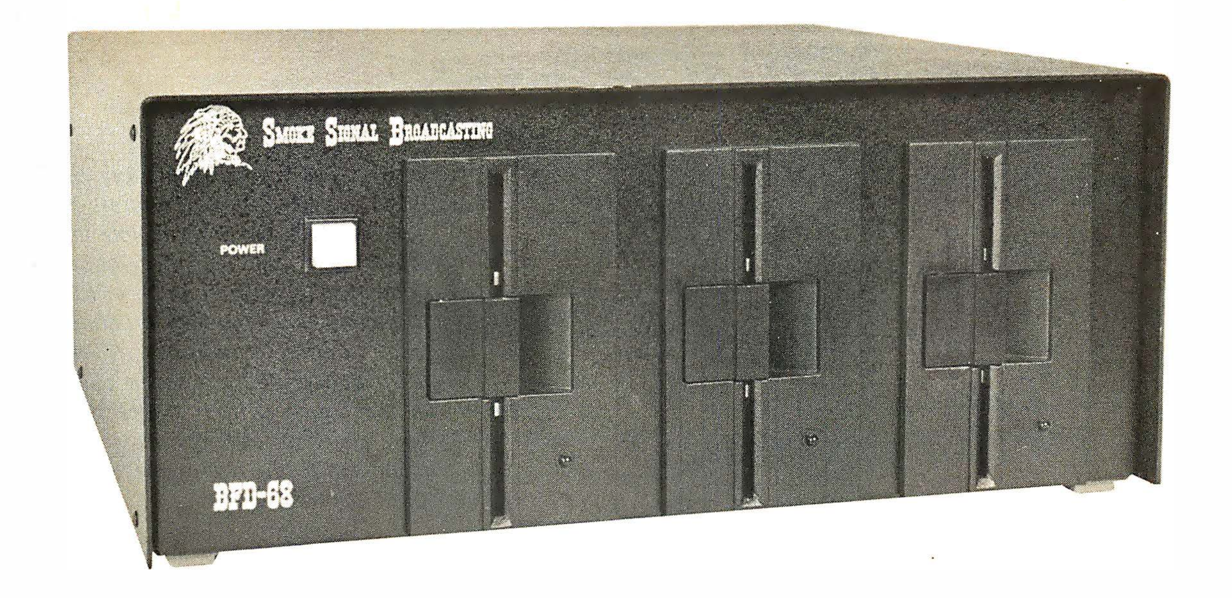


Figure 2.3: Pull up resistors for backplane address lines, and inverting receivers for local use in the front panel interface.



- ANY NUMBER OF FILES MAY BE OPEN (IN USE) AT ONE TIME
- THE NUMBER OF FILES AND SIZE OF FILES IS LIMITED ONLY BY THE SIZE OF THE DISC
- MERGING FILES REQUIRES NO EXTRA DISC SPACE
- NO WAITING FOR THE DISC TO RE-PACK
- LONGER DISC LIFE — MORE EVEN DISC WEAR

Our Basic Floppy Disc System (BFD-68) must, in all modesty be called superb. It comes completely assembled with a disc controller that is plug compatible with the SWTPC 6800. In fact all our products use the 6800 standard SS-50 (Smoke Signal 50) bus used by SWTPC. The cabinet and power supply are capable of handling up to 3 Shugart Mini-Floppy Drives. One drive is included in the price of the BFD-68 and others may be added easily at any time. Or you may save money by ordering the dual-drive BFD-68-2 or triple drive BFD-68-3 (pictured). Price: BFD-68 $795, BFD-68-2 $1139, BFD-68-3 $1479, SA-400 Drive $360.

The BFD-68 includes our Disc Operating System Software. Our software provides for a soft-sectored disc format consisting of 128 bytes per sector, 18 sectors per track and 35 tracks per disc. The software provides direct commands to name and rename files, transfer memory to disc and disc to memory and to automatically jump to the starting location of any program loaded from disc to memory. The direct command names are: RUN, GET, GETHEX, CLOSE, SAVE, DELETE, APPEND, RENAME, COPY, LIST, FIND, LINK and PRINT. In addition, the Disc File Management subroutines are available to create files under your program control.

A bootstrap PROM is included on the controller board to initiate the Disc Operating System which loads into a 4K memory board located at 7000 or optionally at D000. Thus, you can be up and running from a cold start in just a few seconds.

SUPER SOFTWARE

Free patches are provided for SWTPC BASIC version 2.0 and Co-Resident Editor/Assembler. These patches allow the SAVE and LOAD commands to work with the disc or the cassette at your option.

SUPER EDITOR: Smoke Signal Broadcasting now has its own editor. It is a content oriented editor with string search and block move capability. Changes may be made by referring either to line number or string content or a combination of references. Naturally, it is designed for file transfer to and from the BFD-68. Price: SE-1 $29 on diskette or cassette.

ALL OUR PRODUCTS EXCEPT THE PS-1 ARE COMPLETELY ASSEMBLED.

SUPER ASSEMBLER: Inputs source code from file on the BFD-68 disc system and outputs object code to disc file. Assembly listings include alphabetized and tabulated symbol table. Price: SA-1 $29 on diskette or cassette.

Complete source listing included for both editor and assembler. Order both for $53 and save $5.

SMARTBUG — A CURE FOR MIKBUGITIS: A super smart Motorola-Mikbug replacement that preserves almost all Mikbug entry locations so your present programs will run without modification. Uses ACIA instead of PIA and includes many additional features including a software single-step trace command. A SMARTBUG listing is included and object code is provided on a 2708 free with each P-38 series board purchased. Source listing available separately for $19.50.

NEED A FULL SIZE FLOPPY? Our P-38-FF is a plug-in interface card to the ICOM Frugal Floppy™. It includes all the features of the P-38-I plus one 2708 EPROM containing the ICOM bootstrap software. Just plug the P-38-FF into your SWTPC 6800 and your ICOM into the P-38-FF and you're ready to use the Frugal Floppy and ICOM's 6800 software package. Price $299.

Our P-38 is an 8K EPROM board containing room for 8 2708's. Or, you may use it to hold up to 7 2708's plus your Motorola Mikbug or Minibug II ROM. The P-38 addressing is switch selectable to any 8K location. Price $179.

The P-38-I contains all the features of the P-38 plus an interface to the Oliver Paper Tape Reader and our EPROM Programmer. Price $229.

The PS-1 Power Supply Kit provides plus and minus 16 volts required for the P-38 series boards. Also, it allows a wiring modification to be made to the 8 volt supply that will increase its output by one volt. Price $24.95.

Our M-16-A is a 16K single power supply STATIC RAM memory system. The M-16-A is fully buffered and requires only half the power of a similar size system using low-power 2102's. With the M-16-A, you can expand your system to 48K and still have room left for one of our EPROM boards. The M-16-A is switch selectable to any 4K starting address and hardware write protect is included. Quick delivery. Price. $529.

BANKAMERICARD, VISA AND MASTER CHARGE WELCOME.



SMOKE SIGNAL BROADCASTING

P.O. Box 2017, Hollywood, CA 90028 • (213) 462-5652

Circle 128 on inquiry card.

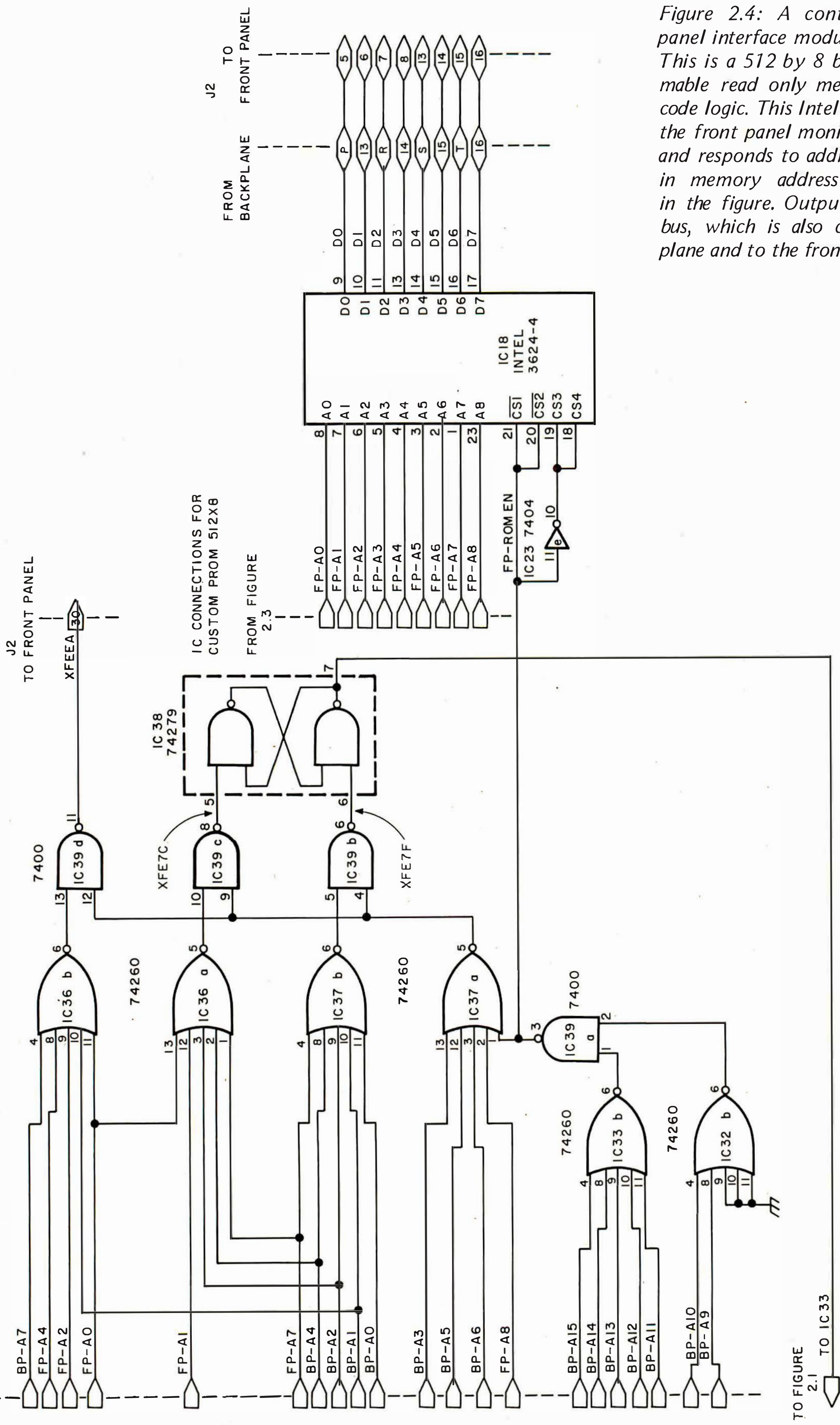
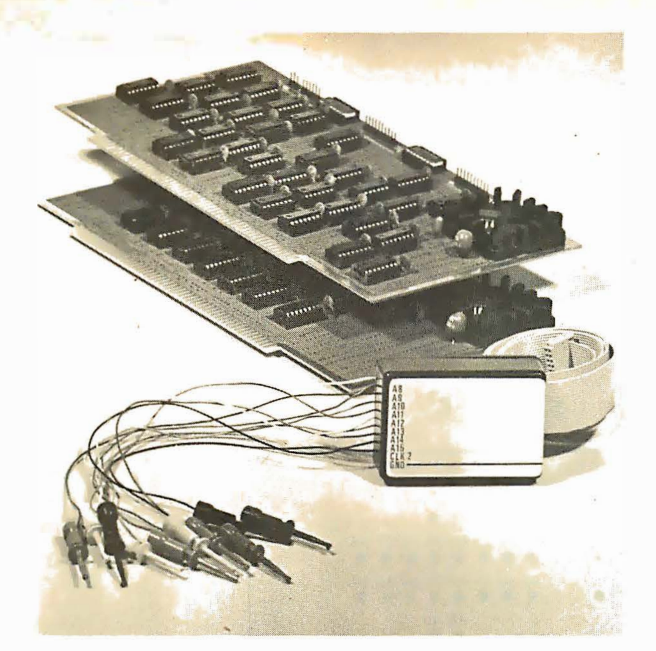


Figure 2.4: A continuation of the front panel interface module address decode logic. This is a 512 by 8 bit fusible link programmable read only memory module with decode logic. This Intel 3624-4 PROM contains the front panel monitor program of listing 1 and responds to addresses FE00 thru FFFF in memory address space using decoding in the figure. Output is directly to the data bus, which is also connected to the backplane and to the front panel assembly.

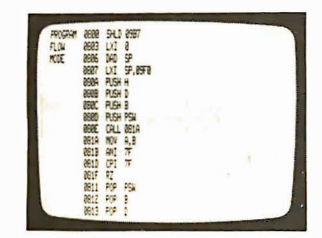


24 Channel **LOGIC ANALYZER**, complete with 2 cards and 3 sets of probes (only one set shown).

24 channel Logic Analyzer plugs into your S-100 Bus

Features

- 24 channels with 256 samples each.
- Display of disassembled program flow.
- Dual mode operation – external mode analyses any external logic system. Internal mode monitors users data and address bus.
- Selectable trigger point anywhere in the 256 samples.
- 0-16 bit trigger word format or external qualifier.
- 10MHz sample rate (50ns min. pulse width)
- Synchronous clock sample with coincident or delayed clock mode.
- User defined reference memory.
- Displays and system control through keyboard entry.
- TTL Logic level compatible (15 pf and 15 μa typical input loading).
- Includes annotated source listing.



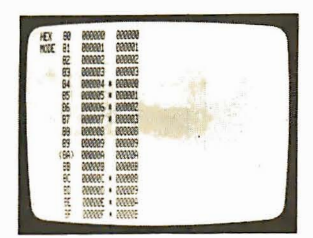
Display of disassembled program flow.

The DATALYZER

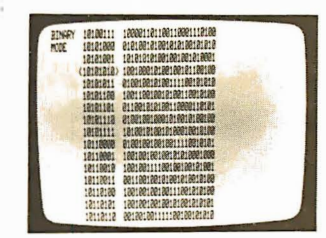
The Databyte Logic Analyzer (DATALYZER) is a convenient, flexible, high quality device. Efficient engineering has allowed a combination of features previously available in only the most expensive units.

Designed to plug easily into your S-100 Bus, the DATALYZER is a complete system –– for only $495. Display of disassembled program flow is a standard feature, not an extra. And the low price includes 30 logic probes, so you can hook up immediately, without additional expense.

The DATALYZER is available in kit form ($495), and as a fully assembled device on two PCB's ($595). Four-week delivery, a substantial warranty, and the Databyte, Inc. commitment to service make the DATALYZER a worthwhile investment. Begin debugging by sending the coupon now.



Displays in Hex



Displays in Binary

Please send me the 24 Channel **LOGIC ANALYZER**
☐ Kit – (manual included) $495.00 (Wis. res. add 4%)
☐ Assembled and Tested (manual included) $595.00
☐ Operators' manual only $7.50
Delivery of all items in four weeks to:
Name ______
Address ______
City ______ State ______ Zip ______
Telephone ______
Payment Enclosed: ☐ Check ☐ Money Order
☐ BankAmericard ☐ Master Charge Exp. Date ______
Number ______
Signature ______

Databyte, Inc.

P.O. Box 14
7433 Hubbard Avenue
Middleton, Wisconsin 53562
Tel: (608) 831-7666

Circle 43 on inquiry card.

Figure 2.5: Mechanical layout of the front panel interface module. This board is built on a standard Vector Electronic Co prototyping card, and plugs into the 44 pin backplane connector (two sides with 22 pins each).

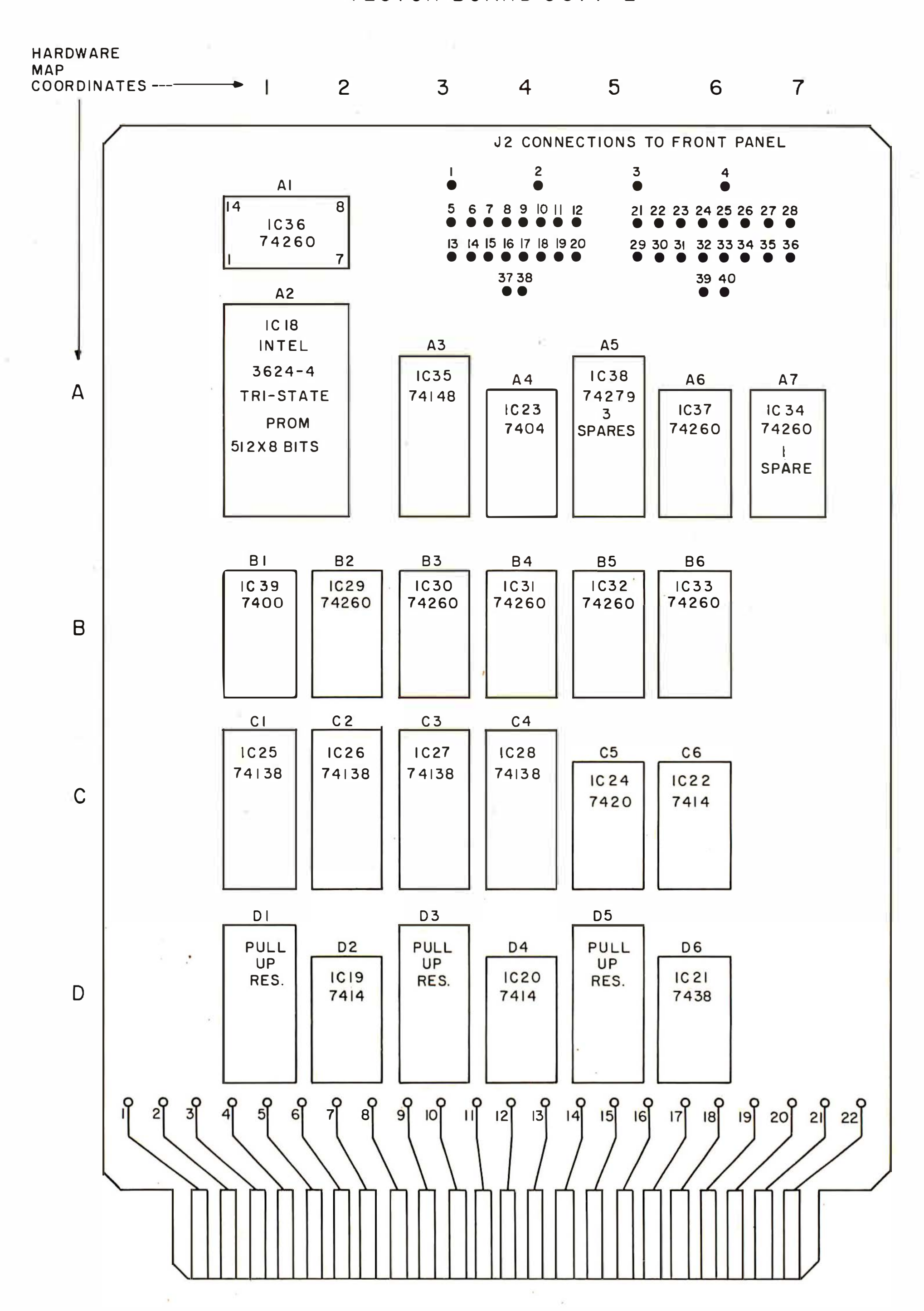
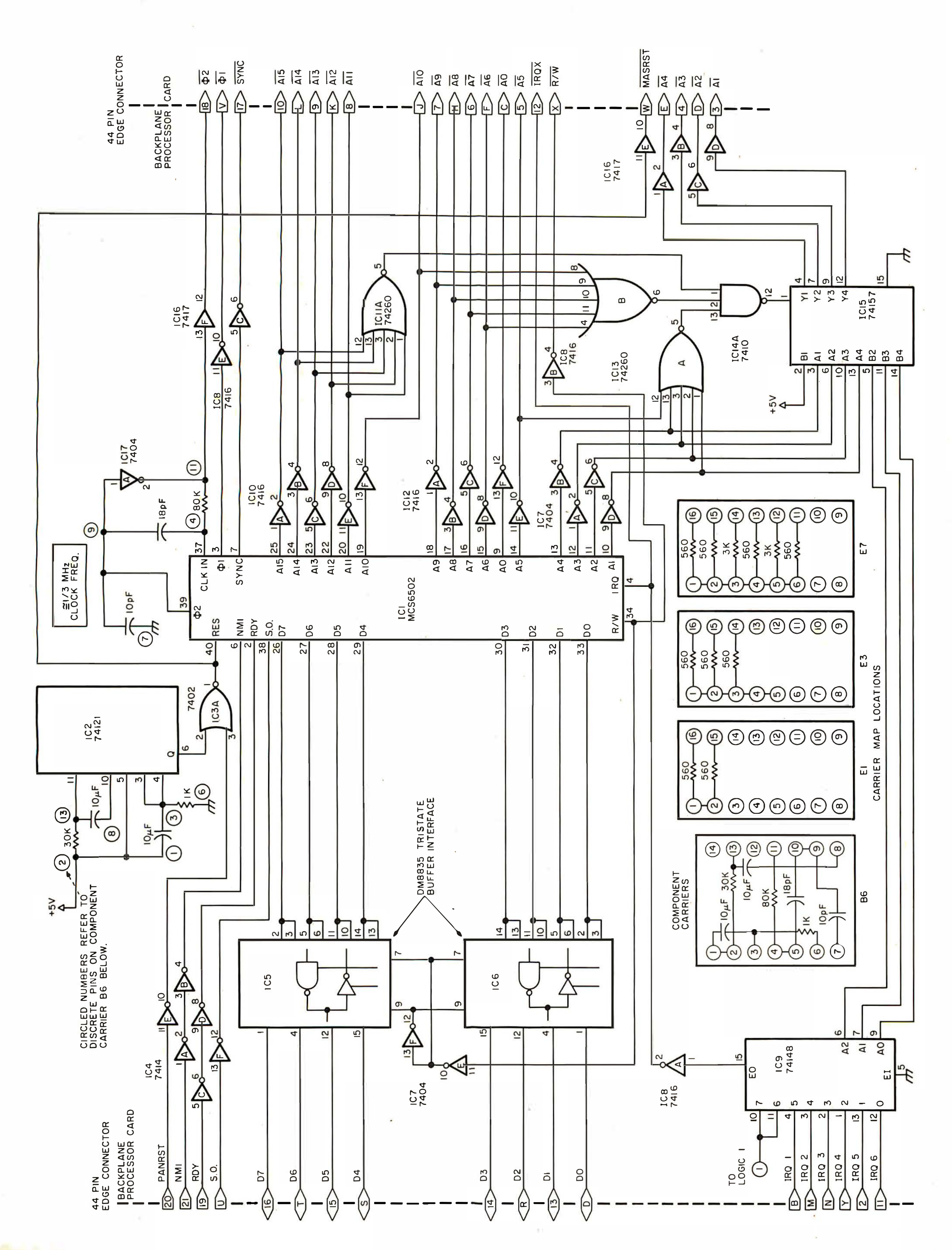


Figure 3.1: Processor module diagram. The 6502 processor module contains the 6502, its data bus and address bus buffering logic, and logic of the priority interrupt structure which substitutes one of six interrupt vector addresses on the low order address lines based on the priority of an interrupt. This overrides the processor generated address of FFFE or FFFF for an interrupt via the IRQ input. IRQ interrupt vectors are located at addresses FFE4 to FFEE in memory address space, with six priority levels. Note that the BRK instruction maps to vector FFE0 (see listing 1).


44 PIN EDGE CONNECTOR
BACKPLANE PROCESSOR CARD
CIRCLED NUMBERS REFER TO DISCRETE PINS ON COMPONENT CARRIER B6 BELOW.
≅1/3 MHz CLOCK FREQ.
IC1 MCS6502
IC2 74121
IC3A 7402
IC4 7414
IC7 7404
IC8 7416
IC10 7416
IC11A 74260
IC12 7416
IC13 74260
IC14A 7410
IC15 74157
IC16 7417
IC17 7404
IC9 74148
DM8835 TRISTATE BUFFER INTERFACE
TO LOGIC 1
COMPONENT CARRIERS
CARRIER MAP LOCATIONS
B6
E1
E3
E7

Text continued from page 122

its rising transition at the end of the pulse, the state of the execution state flip flop changes, causing the halt mode to be entered.

The logic which drives the RDY and NMI lines is responsible for assuring that the processor runs one extra instruction before dropping off into a halt. The process of going into a halt is accomplished through the nonmaskable interrupt. A halting of the user program really means return to the front panel control program through the NMI signal generated here, so that the front panel control program can use the register information stacked up during the interrupt to update the external displays.

After such a halt, the front panel address display shows the location of the next instruction which can be executed in the program just halted. By setting the data display

Text continued on page 134

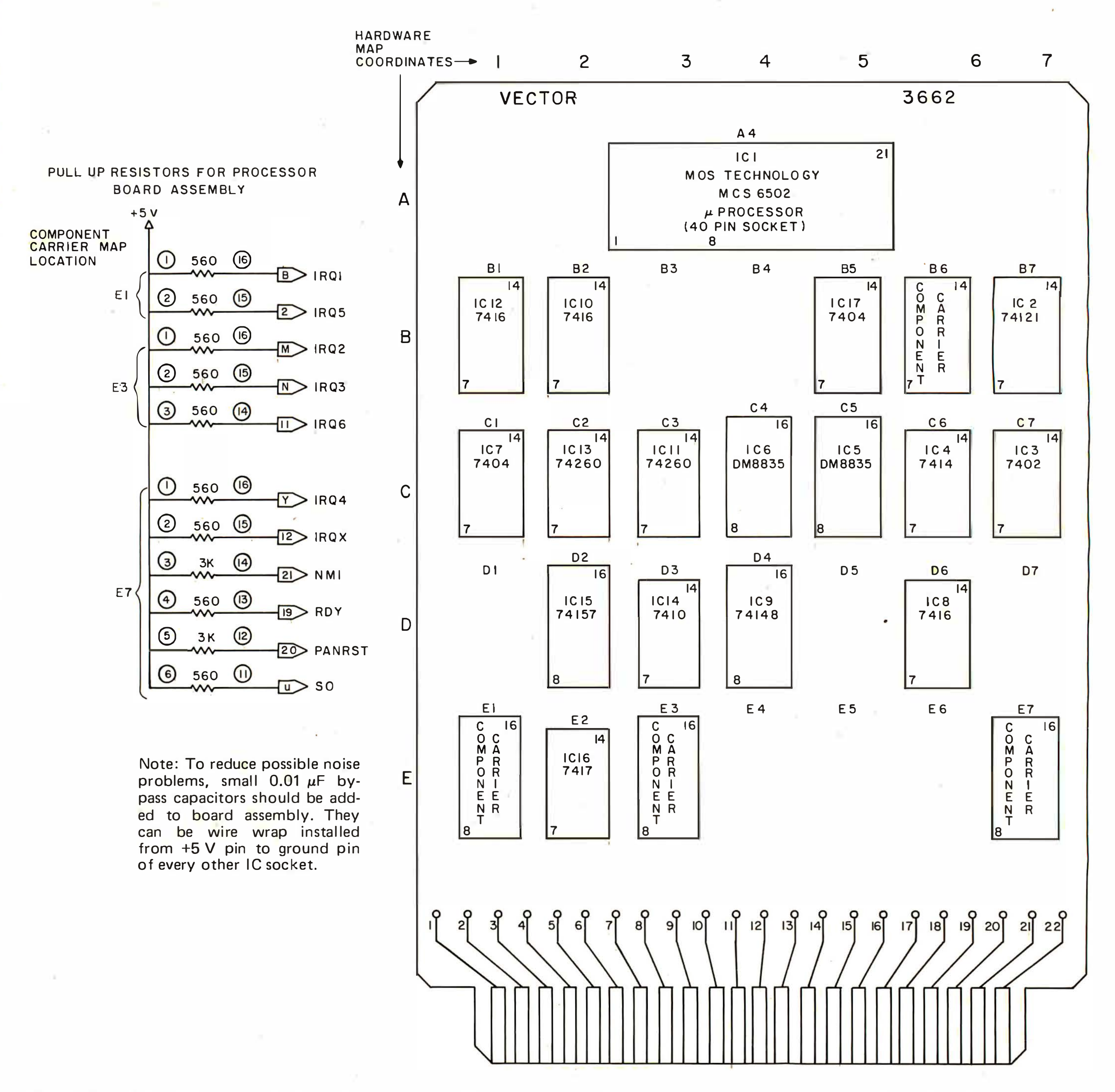
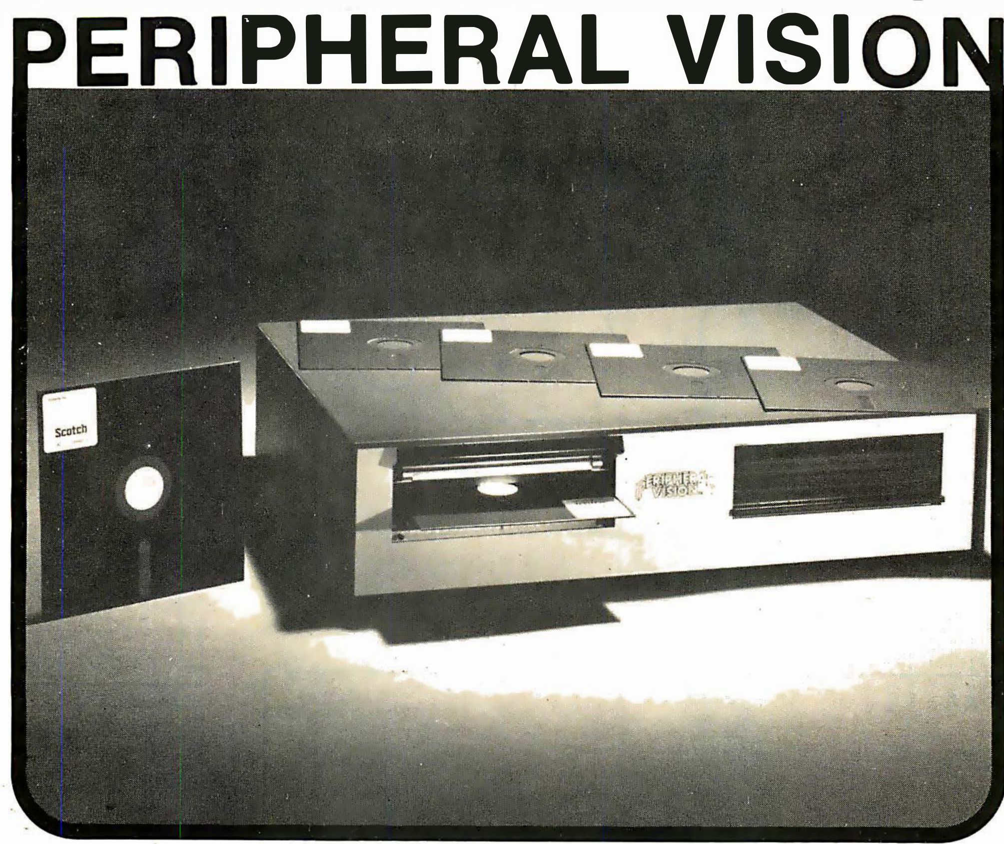


Figure 3.2: Processor module mechanical layout. This map shows placement of the processor integrated circuits on a Vector prototyping card.

take a close look at PERIPHERAL VISION



Peripheral Vision is a young, fast-moving company that's dedicated to selling reasonably priced peripherals for various manufacturers' CPU's.

So now, when you build your microcomputer system, you'll know where to look for all the peripherals that will make your system do what it's supposed to do.

Peripheral Vision may be young, but we have some old-fashioned ideas about how to run our business.

We know there are serious incompatibilities among the various manufacturers' peripherals and CPU's. We want to get them together. And we want to bring significant new products to market--products consisting of everything from adaptation instructions/kits for hardware and software to major new designs.

Most important to our customers, Peripheral Vision is committed to helping you get along with your computer. We'll do all we can to make it easy.

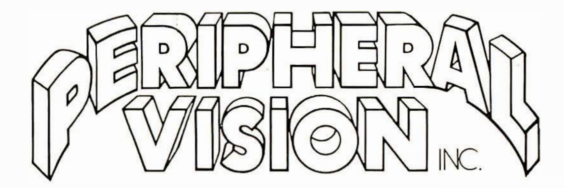
Our first product is a real reflection of this philosophy. It's a full-size floppy disk for the Altair-Imsai plug-in compatible S-100 BUS. And it's available for as low as $750.00.

Our floppy disk has many exciting features:

- 1 interface card supports 4 or more drives
- Stores over 300,000 bytes per floppy
- Bootstrap EPROM included--no more toggling or paper tape
- Completely S-100 plug-in compatible
- Drive is from Innovex (the originator of the floppy concept)--assembled and tested
- Disk operating system with file management system included on floppy
- Cabinet and power supply optional

Also in the works are many new products we'll be letting you know about soon, if you'd like to take a closer look. Like I/O cards, tape drives, an impact printer--all for the S-100 BUS--and we're designing peripherals for a lot of other CPU's too.

We've given you a little glimpse of who we are and what we're doing. If you want to see more, just fill in the coupon below.



P.O. Box 6267/Denver, Colorado 80206 303/777-4292

Name ______________________

Address ______________________

City/State/Zip ______________________

Circle 114 on inquiry card.

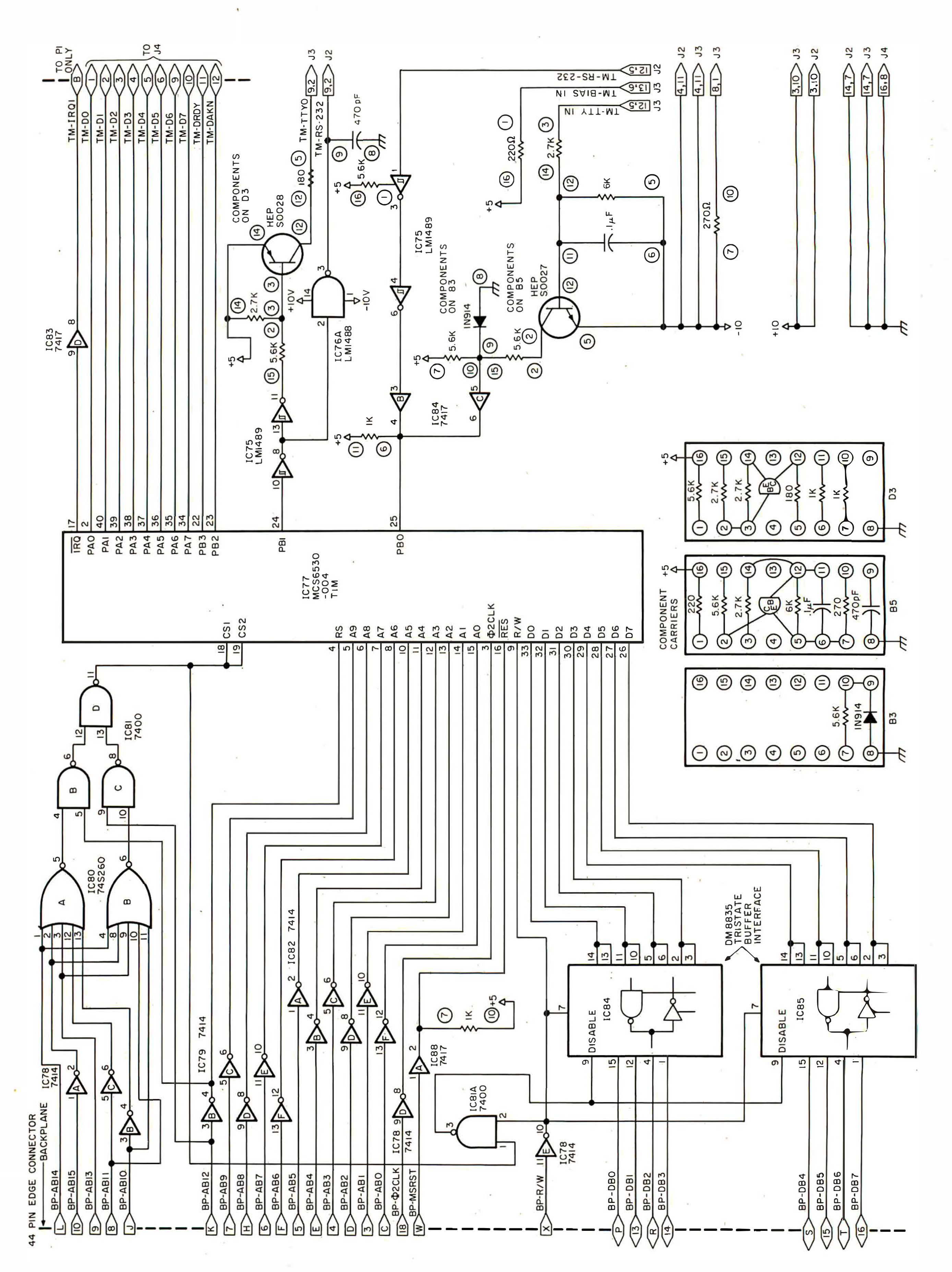
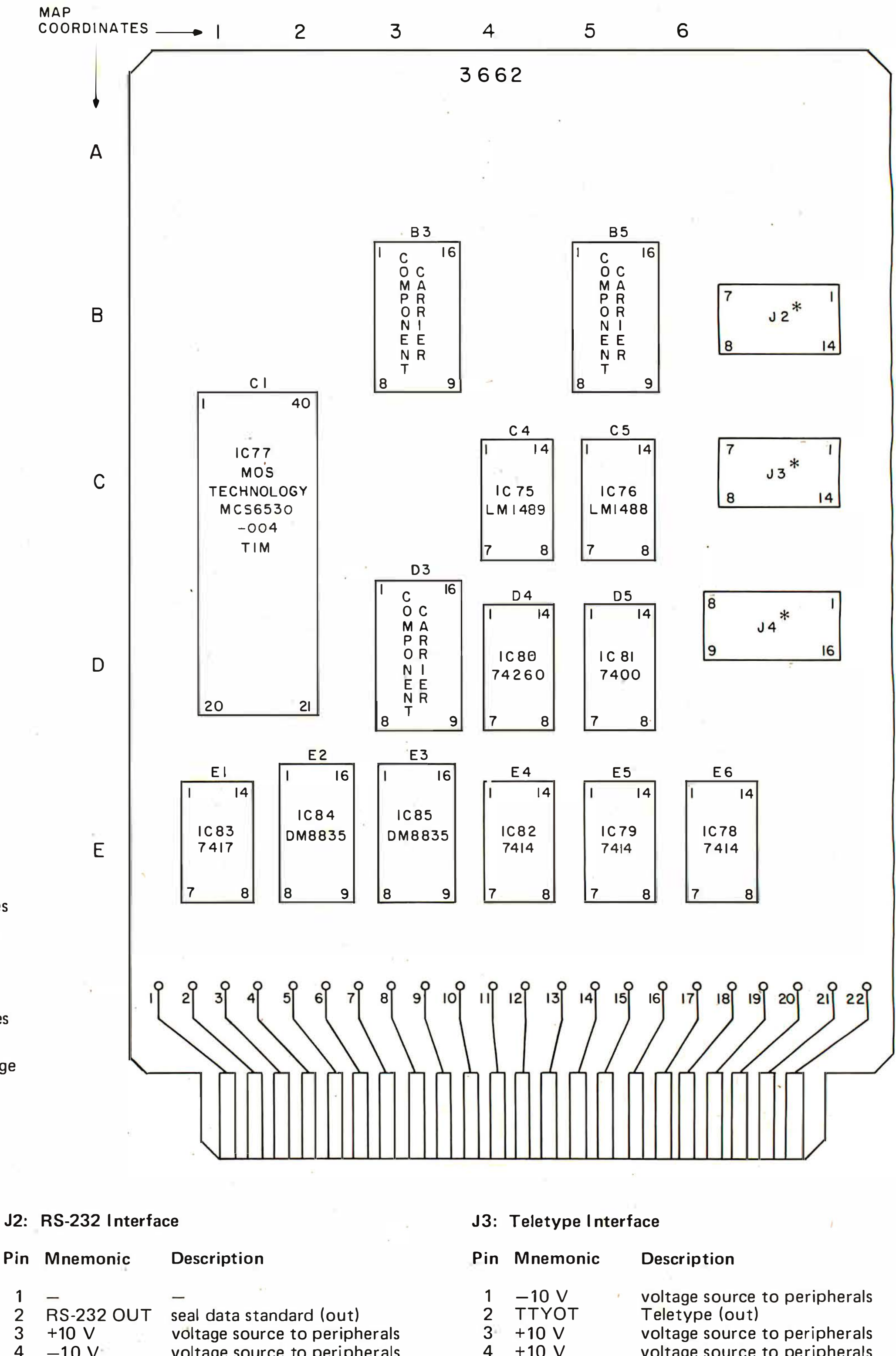


Figure 4.1: TIM interface module details. The MOS Technology TIM monitor program resides in a single MCS6530 ROM and peripheral interface circuit. The TIM interface module allows Kompuutar to be used with any serial terminal.



J4: Parallel Interface

Pin	Mnemonic	Description
1	D0	data bus lines
2	D1	data bus lines
3	D2	data bus lines
4	D3	data bus lines
5	D4	data bus lines
6	D5	data bus lines
7	—	—
8	GND	ground
9	D6	data bus lines
10	D7	data bus lines
11	DRDY	data ready
12	DAKN	data acknowledge
13	—	—
14	—	—
15	—	—
16	GND	ground

*Jacks 2, 3 and 4 are standard IC wire wrap sockets. They are used as cable connectors by mating with special "Augat" plugs.

J2: RS-232 Interface

Pin	Mnemonic	Description
1	—	—
2	RS-232 OUT	seal data standard (out)
3	+10 V	voltage source to peripherals
4	−10 V	voltage source to peripherals
5	RS-232 IN	seal data standard (in)
6	—	—
7	GND	ground
8	—	—
9	RS-232 OUT	seal data standard (out)
10	+10 V	voltage source to peripherals
11	−10 V	voltage source to peripherals
12	RS-232 IN	seal data standard (in)
13	—	—
14	GND	ground

J3: Teletype Interface

Pin	Mnemonic	Description
1	−10 V	voltage source to peripherals
2	TTYOT	Teletype (out)
3	+10 V	voltage source to peripherals
4	+10 V	voltage source to peripherals
5	TTYIN	Teletype (in)
6	BIASIN	pull up voltage source
7	GND	ground
8	−10 V	voltage source to peripherals
9	TTYOT	Teletype (out)
10	+10 V	voltage source to peripherals
11	−10 V	voltage source to peripherals
12	TTYIN	Teletype (in)
13	BIASIN	pull up voltage source
14	GND	ground

Figure 4.2: TIM interface module mechanical layout. This shows the physical arrangement of the wire sockets used to implement the TIM terminal interface for Kompuutar.

Text continued from page 128

selection switch to point to the DATA position, the contents of memory at this location are displayed, having been transferred to the front panel scratch pad location 800D by the service program during the halt operation. At other positions of the switch, it is possible to view copies of the X register, Y register, accumulator A or the stack pointer register. The current contents of the processor status register are shown in the eight discrete LED outputs at location 8013 in memory address space.

If it is desired to modify a status flag, the flag select switch can be rotated to the desired bit, and the flag set or flag reset function be used to alter the flag state. If desired, memory can be examined or altered using the deposit, deposit next, examine or examine next routines activated by appropriate panel switches. Pressing RUN con-

Text continued on page 137

Central Processor Module Integrated Circuits

IC	Type	+5 V	GND	Map Location in Figure 3.2
1	MSC6502	8	1, 21	A3, 4, 5
2	74121	14	7	B7
3	7402	14	7	C7
4	7414	14	7	C6
5	DM8835	16	8	C5
6	DM8835	16	8	C4
7	7404	14	7	C1
8	7416	14	7	D6
9	74148	16	8	D4
10	7416	14	7	B2
11	74260	14	7	C3
12	7416	14	7	B1
13	74260	14	7	C2
14	7410	14	7	D3
15	74157	16	8	D2
16	7417	14	7	E2
17	7404	14	7	B5

Front Panel Interface Module Circuits

IC	Type	+5 V	GND	Map Location in Figure 2.3
18	Intel3624-4	22, 24	12	A2
19	7414	14	7	D2
20	7414	14	7	D4
21	7438	14	7	D6
22	7414	14	7	C6
23	7404	14	7	A4
24	7420	14	7	C5
25	74138	16	8	C1
26	74138	16	8	C2
27	74138	16	8	C3
28	74138	16	8	C4
29	74260	14	7	B2
30	74260	14	7	B3
31	74260	14	7	B4
32	74260	14	7	B5
33	74260	14	7	B6
34	74260	14	7	A7
35	74148	16	8	A3
36	74260	14	7	A1
37	74260	14	7	A6
38	74279	16	8	A5
39	7400	14	7	B1

Front Panel Assembly Circuits

IC	Type	+5 V	GND	Map Location in Figure 1.9
40	74279	16	8	D5
41	74279	16	8	E5
42	74279	16	8	E6
43	74279	16	8	F8
44	7400	14	7	E3
45	74197	14	7	D1
46	7400	14	7	F9
47	7400	14	7	F1
48	7404	14	7	C1
49	7474	14	7	F4
50	74148	16	8	F7
51	7474	14	7	F5
52	7400	14	7	F6
53	7410	14	7	F3
54	7400	14	7	C5
55	7474	14	7	F2
56	74197	14	7	A4
57	7416	14	7	B5
58	7400	14	7	C4
59	74197	14	7	B4
60	7416	14	7	B1
61	7489	16	8	A3
62	74157	16	8	C3
63	7489	16	8	B3
64	7416	14	7	A1
65	HP-5082-7340	7	6	X1
66	HP-5082-7340	7	6	X2
67	HP-5082-7340	7	6	X3
68	HP-5082-7340	7	6	X4
69	74153	16	8	A2
70	74153	16	8	B2
71	74153	16	8	C2
72	74153	16	8	E4
73	DM8835	16	8	E2
74	DM8835	16	8	E1

TIM Interface Module Circuits

IC	Type	+5 V	+10 V	−10 V	GND	Map Location in Figure 4.2
75	LM1489	14	—	—	7	C4
76	LM1488	—	14	1	7	C5
77	MCS6530	20	—	—	1	C1
78	7414	14	—	—	7	E6
79	7404	14	—	—	7	E5
80	74S260	14	—	—	7	D4
81	7400	14	—	—	7	D5
82	7404	14	—	—	7	E4
83	7417	14	—	—	7	E1
84	DM8835	16	—	—	8	E2
85	DM8835	16	—	—	8	E3

Table 5: Integrated circuit summary for Kompuutar. This table summarizes the 85 integrated circuits used in Kompuutar, arranged in groupings by the circuit modules of the system. The column labelled "Map Location" identifies the physical position of the circuits on the various boards of the system, as shown in the physical layout diagrams in the figures. Note that the physical layouts represent a good workable arrangement of sockets. There is no logical requirement that the particular map positions used in these diagrams be followed to the letter in another implementation of the system.

A Mega·Success in 1977!

NOW PREPARING FOR ITS SECOND SHOWING

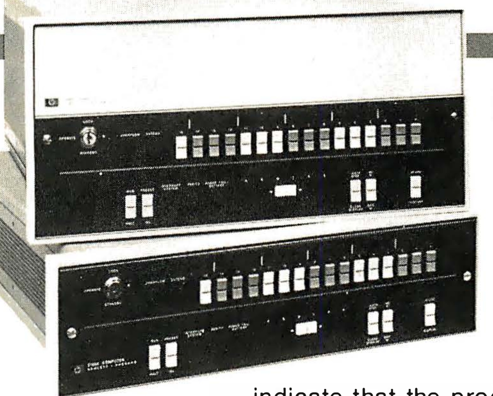
International Microcomputers Minicomputers Microprocessors '78

20, 21, 22 JUNE, 1978

Palais des Expositions Geneva, Switzerland

ANOTHER PRIME OPPORTUNITY TO SELL TO THE EUROPEAN MARKET

Microcomputers, Minicomputers, Microprocessors, Peripherals, Components and Services



A 50% LARGER EXHIBITION IS PLANNED

The sales results IMMM obtained for its 1977 exhibitors is clearly indicative that in 1978 more and more producers will be displaying products for use in every type of industrial, commercial, consumer and military application. Their enthusiasm has prompted many additional manufacturers of small computers (firms which attended and observed in 1977) to make serious commitments regarding participation in the next show. With these new exhibitors and the increased space already requested by 1977 participants, IMMM '78 will be a much larger show!

In 1978, the kind of people you want to meet — executives, engineers, designers, manufacturing and support supervisors, and others — will be out in force . . . to see, to learn, to BUY. And you will want to be there with YOUR products and services.

PROGRAMME DESIGNED TO ATTRACT MANY MORE VISITORS

The remarkably large audience of highly qualified and seriously interested visitors who attended the first IMMM exposition was obviously pleased with the technical programme. Comments indicate that the programme, as well as the exhibition, will be a key factor in attracting an even larger group of attendees to the next show.

The 1978 programme, chaired and presented by internationally recognised experts, again will be designed to offer the kind of practical solutions to day-to-day problems that attendees seek. A special session on "Tips for Hobby Microcomputers" is being planned.

SPONSORED BY DISTINGUISHED PUBLICATIONS

Mini-Micro Systems (U.S.A.) **Polyscope (Switzerland)**
Micomp (Switzerland) **Markt Und Technik (West Germany)**

For Additional Information:

IN EUROPE:
Mr. Bert Saunders
Kiver Communications S.A.
(U.K. Branch Office)
171/185 Ewell Road
Surbiton,
Surrey KT6 6AX England
Telephone: 01-390-0281
Telex: 929837

Mr. Ernest Jungmann
Promotion Marches Exterieurs
Residence Mexico
65, rue du Javelot
75645 Paris CEDEX 13, France
Telephone: (1) 583-96-62
Telex: 210500F

IN THE UNITED STATES:
Mr. Joseph Maurer
Industrial & Scientific Conference Management, Incorporated
222 West Adams Street
Chicago, Illinois 60606
Telephone: (312) 263-4866
Telex: 256148

IN JAPAN:
Mr. K. Yamada
ISCM Japan
Kokado Building
1-3-18 Akasaka
Minato-ku, Tokyo 107, Japan
Telephone: 03-585-8321
Telex: 28887

Circle 66 on inquiry card.

Listing 1: The front panel service program. This program is resident in a programmable read only memory part wired for addresses FE00 to FFFF, as shown in figure 2.4. The listing is a symbolic assembly language version of the program combined with the hand assembled object code provided by author Brader. The information at the end of the PROM area includes the interrupt vectors which are implemented in the Kompuutar system.

```
Hexadecimal Hexadecimal
Address     Code        Label    Op   Operand    Commentary

                        * PANEL DRIVER NONMASKABLE INTERRUPT
                          ENTRY
FE00        8D 00 80    PNMI     STA  SVACC      [Save state
FE03        8E 01 80             STX  SVX         of processor registers
FE06        8C 02 80             STY  SVY         at interrupt] ;
FE09                    * RECOVER PROCESSOR STATUS & RETURN
                          ADDRESS AT INTERRUPT
FE09        68          UNSTK    PLA
FE0A        AA                   TAX
FE0B        68                   PLA
FE0C        A8                   TAY
FE0D        68                   PLA
FE0E                    * SAVE STATUS, RETURN ADDRESS & STACK IN
                          SCRATCH PAD
FE0E        8E 03 80    SAVERET  STX  SVPFLG
FE11        8C 07 80             STY  SVADDR
FE14        8D 08 80             STA  SVADDR+1
FE17        BA                   TSX
FE18        8E 05 80             STX  SVSTK
FE18                    * INTERROGATE CONTROL FUNCTION REQUEST
                          REGISTER
FE18        AD 10 80             LDA  REQST
FE1E        0A                   ASL
FE1F        0A                   ASL
FE20        0A                   ASL
FE21        0A                   ASL
                        * ENTER PROGRAM TREE TO DECODE & EXECUTE
                          REQUEST
FE22        8D 04 80             STA  SVAAA
FE25        A9 20                LDA  #$20
FE27        2C 04 80             BIT  SVAAA
FE2A        10 03                BPL  *+5        If request greater than 7
FE2C        4C 7F FF             JMP  REGLD       then go do register load
                                                  routine;
FE2F        70 1C                BVS  XHSEE      Else if 4 < request < 7
                                                  then halt, step, examine
                                                  or exam next;
FE31        D0 0D                BNE  XDDN       Else if 2 < request < 3
                                                  then deposit or deposit
                                                  next;
                        * CONCLUDE REQUEST IS 0 OR 1
FE33        A9 10                LDA  #$10       [Set up mask for bit
                                                  test] ;
FE35        2C 04 80             BIT  SVAAA
FE38        D0 03                BNE  *+5        Is request 0?
FE3A        4C 99 FE             JMP  EXITCP     If request = 0 then leave
                                                  control program via
                                                  normal NMI;
FE3D        4C 36 FF             JMP  FLAG       If request = 1 then go do
                                                  FLAG function service;
                        * CONCLUDE REQUEST IS 2 OR 3
FE40        A9 10       XDDN     LDA  #$10
FE42        2C 04 80             BIT  SVAAA
FE45        D0 03                BNE  *+5        Is request 2?
FE47        4C 29 FF             JMP  DPSNXT     If request = 2 then go do
                                                  DEPOSIT NEXT
                                                  service;
FE4A        4C 1E FF             JMP  DEPOSIT    If request = 3 then go do
                                                  DEPOSIT service;
FE4D                    * CONCLUDE REQUEST IS 4 TO 7
FE4D        D0 0D       XHSEE    BNE  XHLSS      If request = 6 or request =
                                                  7 then go do HALT or
                                                  STEP;
                        * CONCLUDE REQUEST IS 4 OR 5
FE4F        A9 10                LDA  #$10
FE51        2C 04 80             BIT  SVAAA
FE54        D0 03                BNE  *+5        Is request 4?
FE56        4C FD FE             JMP  EXMNXT     If request = 4 then go do
                                                  EXAMINE NEXT;
FE59        4C EE FE             JMP  EXAMINE    If request =   then go do
                                                  EXAMINE;
                        * CONCLUDE REQUEST IS 6 OR 7
FE5C        A9 10       XHLSS    LDA  #$10       [Set up mask for bit
                                                  test] ;
FE5E        2C 04 80             BIT  SVAAA      [Test for odd or even
                                                  number] ;
FE61        F0 14                BNE  SETTRAP    If request = 7 then skip
                                                  setup logic;
                        * CONCLUDE REQUEST IS 6
FE63                    * SET UP SCRATCH PAD INSTRUCTION SEQUENCE
FE63        A9 8D                LDA  #$8D
FE65        8D 09 80             STA  DUMMY2
FE68        A9 0D                LDA  #$0D
FE6A        8D 0A 80             STA  DUMMY3
FE6D        A9 80                LDA  #$80
FE6F        8D 08 80             STA  DUMMY4
FE72        A9 60                LDA  #$60
FE74        8D 0C 80             STA  DUMMY5
FE77                    * SET UP SCRATCH PAD INSTRUCTION SEQUENCE
FE77        A9 AD       SETTRAP  LDA  #$AD       [Dummy in an LDA
FE79        8D 06 80             STA  DATATRAP    instruction using cur-
                                                  rent address field as its
                                                  address] ;
FE7C                    * ROUTINE TO EXECUTE SCRATCH PAD PROGRAM
                          TO SET UP LATEST SVDATA VALUE
FE7C        20 06 80    SETUP    JSR  DATATRAP   [Call scratch pad resident
                                                  subroutine] ;
FE7F        AD 03 80             LDA  SVPLFG
FE82        8D 13 80             STA  STATUS
FE85        A9 0D                LDA  #$0D       [Patch the scratch pad
FE87        8D 0A 80             STA  DUMMY3      program] ;
FE8A                    * DISPLAY THE CURRENT ADDRESS
FE8A        AD 07 80             LDA  SVADDR
FE8D        8D 14 80             STA  LADDR
FE90        AD 08 80             LDA  SVADDR+1
FE93        8D 15 80             STA  HADDR
FE96        4C 9D FE             JMP  CONTINUE
FE99                    * CONTROL PANEL MONITOR EXIT POINT
FE99        38          EXITCP   SEC             [Set carry to indicate end
                                                  of monitor] ;
FE9A        4C 9E FE             JMP  SETSTAK
FE9D        18          CONTINUE CLC             [Clear carry to continue
                                                  monitor] ;
FE9E                    * BEGIN ROUTINE TO SET UP STACK AND
                          TEMPORARILY LEAVE MONITOR
FE9E        AD 08 80    SETSTAK  LDA  SVADDR+1
FEA1        48                   PHA
FEA2        AD 07 80             LDA  SVADDR
FEA5        48                   PHA
FEA6        AD 03 80             LDA  SVPFLG
FEA9        48                   PHA             [Push processor status
                                                  onto stack] ;
FEAA        AE 05 80             LDX  SVSTK      [Subtract 3
FEAD        CA                   DEX              from
FEAE        CA                   DEX              old
FEAF        CA                   DEX              stack pointer
FEB0        9A                   TXS              to compensate pushes] ;
                        * RESTORE REGISTERS
FEB1        AE 01 80             LDX  SVX
FEB4        AC 02 80             LDX  SVY
FEB7        AD 00 80             LDA  SVACC
FEBA        8D 0E 80             STA  LSTACC
FEBD        B0 3C                BCS  LEAVE      If carry set, then leave
                                                  monitor now;
FEBF                    * MANIPULATIONS FOR SINGLE STEP CASE
                          FOLLOW
FEBF        A9 FF                LDA  #$FF
FEC1        4D 00 80             EOR  SVACC      [Invert old accumulator
                                                  value] ;
FEC4        8D 00 80             STA  SVACC
FEC7        A9 FF                LDA  #$FF
FEC9        4D 01 80             EOR  SVX        [Invert old X value] ;
FECC        8D 01 80             STA  SVX
FECF        A9 FF                LDA  #$FF
FED1        4D 02 80             EOR  SVY        [Invert old Y value] ;
FED4        8D 02 80             STA  SVY
FED7        A9 FF                LDA  #$FF
FED9        4D 05 80             EOR  SVSTK      [Invert old stack register
                                                  value] ;
FEDC        8D 05 80             STA  SVSTK
FEDF        A9 FF                LDA  #$FF
FEE1        4D 0D 80             EOR  SVDATA     [Invert old data register
                                                  value] ;
FEE4        8D 0D 80             STA  SVDATA
FEE7        AD 0E 80             LDA  LSTACC     [Restore old accumulator
                                                  value] ;
FEEA        40                   RTI             [Return from interrupt,
                                                  exiting panel service
                                                  program] ;
FEEB                    * LEAVE PANEL SERVICE FOR USER NMI SERVICE
                          CASE
FEEB        4C 00 F6    LEAVE    JMP  USRNMI
FEEE                    * NOTE: USRNMI IS A READ ONLY MEMORY
                          ROUTINE AT F600, NOT DEFINED HERE
FEEE                    *
FEEE                    *        EXAMINE SERVICE ROUTINE
FEEE        AD 11 80    EXAMINE  LDA  DSWLOW
FEF1        8D 07 80             STA  SVADDR
FEF4        AD 12 80             LDA  DSWHIGH
FEF7        8D 08 80             STA  SVADDR+1
FEFA        4C 77 FE             JMP  SETTRAP    [Go back to display
                                                  routine and exit] ;
FEFD                    *        EXAMINE NEXT SERVICE ROUTINE
FEFD        A0 00       EXMNXT   LDY  #0
                        * INCREMENT TO NEXT ADDRESS
FEFF        AE 07 80    ADVANCE  LDX  SVADDR
FF02        E8                   INX
FF03        8E 07 80             STX  SVADDR
FF06                    * CHECK FOR OVERFLOW
FF06        E0 00                CPX  #0
FF08        F0 0A                BEQ  RAISHI     If low order = 0 then
                                                  increment high order;
FF0A                    * CHECK TO SEE IF INITIALIZATION NEEDED
FF0A        C0 00       QNEEDINI CPY  #0
FF0C        F0 03                BEQ  *+5
FF0E        4C 7C FE             JMP  SETUP      If Y = 0 then simply
                                                  execute scratch pad
                                                  program;
FF11        4C 77 FE             JMP  SETTRAP    If Y = 0 then set up prior
                                                  to executing again;
FF14                    * INCREMENT HIGH ORDER BYTE OF CURRENT
                          ADDRESS
FF14        AE 08 80    RAISHI   LDX  SVADDR+1
FF17        E8                   INX
FF18        8E 08 80             STX  SVADDR+1
FF1B        4C 0A FF             JMP  QNEEDINI
FF1E                    *
FF1E                    * DEPOSIT SERVICE ROUTINE
FF1E        A9 8D       DEPOSIT  LDA  #$8D       [Change scratch pad pro-
FF20        8D 06 80             STA  DATATRAP    gram to store accumu-
                                                  lator (STA) as its first
                                                  operation] ;
FF23        AD 11 80             LDA  DSWLOW
FF26        4C 7C FE             JMP  SETUP
FF29                    *
FF29                    * DEPOSIT NEXT SERVICE ROUTINE
FF29        A9 8D       DPSNXT   LDA  #$8D       [Change scratch pad pro-
FF2B        8D 06 80             STA  DATATRAP    gram to store accumu-
                                                  lator (STA) as its first
                                                  operation] ;
FF2E        AD 11 80             LDA  DSWLOW
FF31        A0 AA                LDY  #$AA       [Load Y with any nonzero
                                                  value, AA in particular] ;
```

Text continued from page 134

tinues execution of the program at the currently displayed address.

After building the design, I found a couple of subtle points in the operation of the control panel. The first point to note is that when using the RESET switch, the processor must be in the RUN mode of the RUN versus STEP switch. Second, bit 5 of the status display is useless and unimplemented in the 6502 hardware. The register load switch (REG LOAD) is best used for modifying the contents of the accumulator, index registers, but not the stack pointer. Using the register load switch to modify the stack pointer can result in problems when resuming execution of a program. Once whatever memory loading or register alteration chores required by a program have been accomplished, execution can be resumed using the RUN switch to cause the control panel program to return from interrupt using the register contents stored in the control panel scratch area.

A Versatile Configuration

The design of Kompuutar is quite readily adaptable to the 6800 processor as a substitute for the 6502 if personal programming preferences or availability of chips dictates such a switch. The similarities between the two processors are quite extensive, and in fact were the bone of contention of a lawsuit (since settled) shortly after the 6502 came out. At the system level, here are the major differences to be aware of:

- *The pinouts of the 40 pin package used for each processor are different, but the signal definitions of NMI, data bus lines, IRQ, reset, address bus, etc, are equivalent.*
- *The 6800 uses four read only memory interrupt vectors at addresses FFF8 to FFFF in memory address space, whereas the 6502 uses only three interrupt vectors; the definitions of the interrupt vectors for reset, nonmaskable interrupt and maskable interrupts are similar.*
- *The instruction sets differ, so the front panel service programs shown with this article would need to be recoded if a 6800 is used.*
- *The definition of the clock used by the processor differs in the details of its drive circuit.*

The major features of either a 6502 or 6800 system at the level of the backplane bus defined here would be nearly identical. ■

Listing 1, continued:

```
Hexadecimal  Hexadecimal
 Address        Code       Label      Op   Operand    Commentary

FF33        4C  FF  FE                JMP  ADVANCE
FF36                      *
FF36                      * FLAG REGISTER CHANGE SERVICE ROUTINE
FF36        AD  10  80    FLAG        LDA  REQST
FF39                      * ALIGN THE FLAG SELECTION BITS WITH FOUR
                            RIGHT SHIFTS
FF39        4A                        LSR
FF3A        4A                        LSR
FF3B        4A                        LSR
FF3C        4A                        LSR
FF3D        8D  04  80                STA  SVAAA
FF40        A9  01                    LDA  #$1
FF42        2C  04  80                BIT  SVAAA      Is flag data bit on?
FF45        F0  05                    BEQ  SETYZER    If flag data = 0 then
                                                        Y:= 0;
FF47        A0  FF                    LDY  #$FF       Else Y:= $FF;
FF49        4C  4E  FF                JMP  PROCFLAG
FF4C        A0  00        SETYZER     LDY  #$00
FF4E        4E  04  80    PROCFLAG    LSR  SVAAA      [Final right shift aligns
                                                        the 3 bit flag code];
FF51        A2  07                    LDX  #$07       [Load loop count];
                          * LOOP SETS X
FF53        EC  04  80    FLGLOOP     CPX  SVAAA      If SVAAA = X then
FF56        F0  04                    BEQ  FCHANGE      go change this flag;
FF58        CA            CYCLE       DEX
FF59        4C  53  FF                JMP  FLGLOOP
FF5C        A9  01        FCHANGE     LDA  #$01
                          * LOOP ALIGNS MASK ON PROPER FLAG BIT
FF5E        E0  00        FLOOK       CPX  #$00
FF60        F0  05                    BEQ  *+7
FF62        0A                        ASL
FF63        CA                        DEX
FF64        4C  5E  FF                JMP  FLOOK
FF67                      * AT END OF LOOP WITH MASK IN PLACE, CHANGE
                            FLAG BIT
FF67        C0  00                    CPY  #$0        Is data 0?
FF69        D0  0B                    BNE  SETFLG     If not then go set flag;
FF6B        49  FF                    EOR  #$FF
FF6D        2D  03  80                AND  SVPFLG     [Turn off the flag bit
                                                        with inverted mask];
FF70        8D  03  80                STA  SVPFLG
FF73        4C  77  FE                JMP  SETTRAP
FF76        0D  03  80    SETFLG      ORA  SVPFLG     [Turn on the selected bit
                                                        from mask];
FF79        8D  03  80                STA  SVPFLG
FF7C        4C  77  FE                JMP  SETTRAP
FF7F                      * REGISTER LOAD SERVICE ROUTINE
FF7F                      * REGISTER LOAD SERVICE ROUTINE
FF7F        A9  0F        REGLD       LDA  #$0F
FF81        8D  0A  80                STA  DUMMY3
FF84        A9  AE                    LDA  #$AE       [Define LDX as first
                                                        operation of scratch pad
                                                        program];
FF86        8D  06  80                STA  DATATRAP
FF89        A9  FF                    LDA  #$FF
FF8B        4D  0D  80                EOR  SVDATA     [Invert data];
FF8E        8D  0D  80                STA  SVDATA
FF91        AD  11  80                LDA  DSWLOW
FF94        4C  7C  FE                JMP  SETUP
- - - - - - - - - - - - - unused space - - - - - - - - - - - - -
FFA0        A2  FF        RESET       LDX  #$FF       [Initialize stack
FFA2        9A                        TXS               pointer];
FFA3        4C  06  70                JMP  TIM        [Jump to TIM monitor
                                                        on system RESET];
- - - - - - - - - - - - - unused space - - - - - - - - - - - - -
FFE0        00  F0        BRK Instruction vector
FFE2        XX  XX        Not used
FFE4        00  FC        IRQ6 Lowest priority
FFE6        00  FB        IRQ5
FFE8        00  FA        IRQ4
FFEA        00  F9        IRQ3 Interrupt vectors
FFEC        00  F8        IRQ2
FFEE        00  F7        IRQ1 Highest priority

- - - - - - - - - - - - - unused space - - - - - - - - - - - - -
FFFA        00  FE        NMI interrupt vector location
FFFC        A0  FF        RESET interrupt vector location
FFFE        XX  XX
```

Kompuutar Peripheral and Scratch Pad Data Symbols:

Address	Symbol	Description
8000	SVACC	Saved accumulator value, scratch pad
8001	SVX	Saved X index register value, scratch pad
8002	SVY	Saved Y index register value, scratch pad
8003	SVPFLG	Saved processor flag register value, scratch pad
8004	SVAAA	Scratch value
8005	SVSTK	Saved stack register value
8006	DATATRAP	First operation of scratch pad program (absolute addressing through SVADDR)
8007	SVADDR	Low order saved address value, scratch pad program absolute address for DATATRAP
8008	SVADDR+1	High order saved address value
8009	DUMMY2	
800A	DUMMY3	Balance of scratch pad program
800B	DUMMY4	
800C	DUMMY5	
800D	SVDATA	Data value at current SVADDR
800E	LSTACC	Accumulator storage temporary
800F	SVAAA	Scratch value
8010	REQST	Front panel request input word
8011	DSWLOW	Low order data switch register input (data or address information)
8012	DSWHIGH	High order data switch register input (address information only)
8013	STATUS	Status lamps output
8014	LADDR	Low order address display output
8015	HADDR	High order address display output

Continued from page 6

the "Thorsen Memory Tube," a magical device which can be programmed with the data necessary for controlling an interactive mechanism. In the real world, technology has developed the semiconductor memory and microprocessors, charge coupled devices and magnetic bubble memories.

In the fictional account, various societal conditions have led to a predilection for garage shop and cottage industry experimentation represented by the efforts of the hero of the story, Dan Davis. In the real world, the technological and economic conditions of today have made the same sort of individualized experimentation with the high technology of computers an everyday occurrence practiced by large numbers of people.

In the story, the protagonists face a world in which mass-produced robotic mechanisms for domestic use have yet to be produced. In the real world, we face a similar situation in which the first mass-produced intelligent robotic mechanisms for domestic use are on the threshold of invention and possibility.

With this possibility of innovation and invention in robotics growing out of the computer experimenter's natural inclination towards artificial intelligence and robotics work, I can begin to build the concept of an ideal type, "The Compleat Robotics Experimenter" and what it takes to become one. Perhaps we'll see a few examples in real life as the next few years roll by.

Categories of Experimentation

In order to put a finger on the categories of experimentation in the general field of robotics, we need a definition of just what is meant by the concept of a robot. I propose the following definition as a working concept for purposes of discussion. This is not necessarily the ultimate definition of what a robot is, but it serves as a standard of measurement useful in this context.

> A robotic system is an intelligent mechanism which is mechanically mobile and which operates with feedback from sensors in a specific environment with general but well defined behavioral goals.

With this definition and given our current levels of technology, I am explicitly attempting to eliminate the classical general purpose robot of science fiction from the discussion. The characters C3PO and R2D2 of George Lucas' film *Star Wars* may be use-

ful as long term developmental goals, but current technology just does not support practical implementation of such delightful pseudo-persons. The kind of robotic mechanism which is likely to be realizable in the near future by real world personal computing experimenters will be more specialized.

The "intelligent" of intelligent mechanism in the definition might be better expressed as "intelligently designed," for it is the crystallization of the designer's intelligence and creativity in the control algorithms of the machine which enables the robotic mechanism to act "intelligently." "Mechanically mobile" in the definition could be as simple as the mobility of the end of a simple arm mechanism, as complicated as the three-dimensional mobility of a remote flying and hovering mechanism, or as conventional as the rolling mobility of Ralph Hollis' robot NEWT. Sensory feedback is essential to the definition, for I wish to exclude from discussion such conventional mechanisms as plotters, printers, and mass storage devices which use limited forms of mechanical mobility.

A "specific environment" is essential to the practicality of the concept if it is to be accomplished at current levels of understanding of artificial intelligence research and engineering.

A quite practical system for the personal computing experimenter with a mechanical flair is the construction of an "arm" mechanism to act as an output device for a chess program. But the practicality of the possibility comes from the limitation of the environment to a three-dimensional region of space above a well defined chess board, with chess pieces designed to fit the design of the arm's grasping mechanisms and object sensors.

Similarly, the person designing the robotic vacuum cleaner appliance can initially implement a practical design only by limiting its environment, requiring that it:

- Avoid precipices (as at the top of stairs).
- Roll on a plane surface of normal gravity.
- Bounce off walls and furniture.
- Only swallow items smaller than a critical size, while sorting and classifying loose objects above that size but below a maximum size.
- Sense presence of animals, children and adults as a cue to enter standby mode of operation.

The environment of the first such practical

household robot would specifically exclude any attention to global details such as what to do when it gets wet, what to do in case of fire, how to wash dishes, how to teach children symbolic logic, etc. (Incidentally, the vacuum cleaner just described is one of the robotic mechanisms designed by the hero of Robert Heinlein's *Door Into Summer*.)

The behavioral design of the robot is a part of this process of limiting the environment. By deciding what the robot is to do, as in the vacuum cleaner case or the chess playing arm case, we impart constraints on its behavior.

The strongest theme of research and experimentation with robotic mechanisms for the near future, then, is that of picking and choosing a particular flavor of environment in which the system is to operate, using this environment definition as the evaluation standard for the design. The environment chosen is what drives the design of mechanisms to interact with that environment; image and sensor processing needs of the system in that environment; software requirements of the computers which implement the processing; and possible avenues of exploration for application concepts.

Defining the Compleat Robotics Experimenter

The personal computer experimenter is well on the way to becoming a robotics experimenter. The intellectual challenge of the robot is a step up in abstraction and difficulty, as well as a step up in fascination and unknowns. What are the requirements of a person who expects to achieve measurable results from experimentation with robotics?

First and foremost, the compleat robotics experimenter is well rounded and virtually the classical Renaissance man. A narrow specialization in one particular aspect of computer science, machine design, etc, may be a useful attribute to possess, but to implement comprehensive systems, comprehensive knowledge is required. For the person just beginning formal education at a college level, a combination of liberal arts and philosophy with computer science, physics, biology and engineering is what I would consider a necessary groundwork for later work in robotics. Getting to more specific details, here are some areas of study with reasons for their usefulness to the aspiring robotics experimenter:

Philosophy, particularly epistemology, is crucial. Epistemology is the philosophical discipline concerned with the question "how do we know what we know?" A

practical understanding of epistemology is a necessary starting point for anyone who would design a knowledge oriented or "artificially intelligent" system.

Related to epistemology is the necessity for a thorough and practical understanding of the mathematical basis for programming and computer science: concepts of logic, proof of theorems, organization of knowledge, etc, form a background for much work with computers.

Progressing to more specific technical areas, the aspiring roboticist must obtain a mastery of computer science, natural and artificial languages, information theory, electrical engineering (at the level of utilizing "black boxes" of function), mechanical engineering and biology or physiology. The knowledge of computer science is a must, for no robot is possible without a computer to implement its intelligence. Natural and artificial language understanding is a requirement for any form of high level command and control structures to be built into the software of the robotic system. Information theory and its attendant discussions of the possibility of error and strategies for coping with error is essential. Electrical engineering and mechanical engineering are obvious for design of interactive real world mechanisms. Biology, particularly the physiology of natural mechanisms, is essential background information to this process of robotic mechanism design: not necessarily for its value as a direct model, but certainly for its inspirational value and value as a source of detail ideas about possible approaches which might work out in robotic mechanisms. Thorough familiarity with current technologies is a must as well.

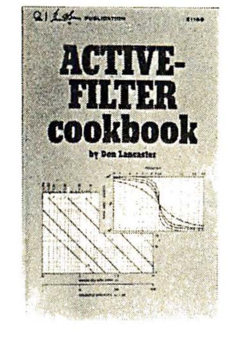
A final requisite is a thorough familiarity with science fiction literature, for its imagination inputs. The science fiction writer is at once a frustrated engineer and a daring source of imagination. The frustrated engineer aspect comes from the lack of a technological context to fullfill the imaginations; the imagination side is the reason why science fiction is a necessary input for the compleat robotics experimenter. Many design ideas can be found in the writings of science fiction thinkers.

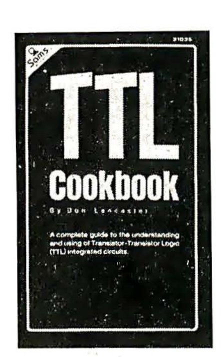
Summing it all up, the personal experimentation with technological concepts which so characterizes the amateur computer person finds a natural extension in the application area of robotics. The challenges and problems of building "smart" machines at once provides a form of an answer to the traditional "what do you do with your computer?" question and a fascinating area for exploration.■

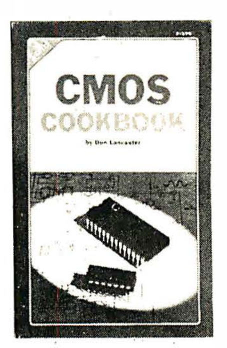
What's Cooking?

. . . The Lancaster Series!









____**TV Typewriter Cookbook** by Don Lancaster. A complete guide to low cost television display of alphanumeric data, several chapters of which were published ahead of the book in early issues of BYTE magazine. $9.95.

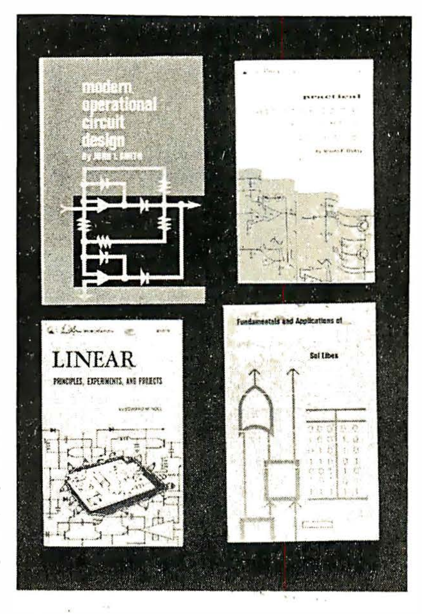
____**Active Filter Cookbook** by Don Lancaster. The chief chef of electronics Cookbooks concocts another gourmet appetizer. Run to this book when you need to find a starting point for the design of a filter for use in an electronic application. $14.95.

____**The TTL Cookbook by Don** Lancaster. Start here with Don's tutorial explanations of what makes TTL logic design tick. 335 pages, $8.95.

____**CMOS Cookbook.** Don Lancaster has taken his highly successful TTL Cookbook formula and done it again. This bigger than ever (414 pages!) book tells you how to use CMOS, the exciting new power logic family. $9.95.

____**Modern Operational Circuit Design** by John L Smith. An absolutely essential introduction to the use and application of operational amplifier systems. The book contains both theoretical background information and practical circuit suggestions which can be used to advantage by the experimenter. $23.25 (hardbound).

____**Linear IC Principles, Experiments, and Projects** by Edward M Noll. From basic principles to complicated systems, from simple amplifier experiments to applications in radio, TV and control systems, this book can improve your knowledge of the way circuitry of the analog world really works. $8.95.



____**Practical Solid-State Circuit Design** by Jerome E Oleksy. A self study course in the design of semiconductor circuits from the simple transistor to the complex operational amplifier. $5.95.

____**Fundamentals and Applications of Digital Logic Circuits** by Sol Libes. An invaluable tutorial background volume on digital logic, arithmetic, IO concepts and interfacing to analog devices; written by one of the founders of the Amateur Computer Group of NJ. This book acquaints the reader with much of the terminology and background concepts of digital hardware. $6.95.

Send to:

BITS, Inc.
70 Main Street
Peterborough NH 03458

master charge
BANKAMERICARD

Check Payment method:
____My check is enclosed
____Bill my MC No. ____________ Exp. date ______
____Bill my BAC No. ____________ Exp. date ______

Name ____________________
Address ____________________
City ________ State ______ Zip Code ______
Signature ____________________

Total for all books checked $______
Postage 50 cents per book for ____ books $______
Grand Total $______

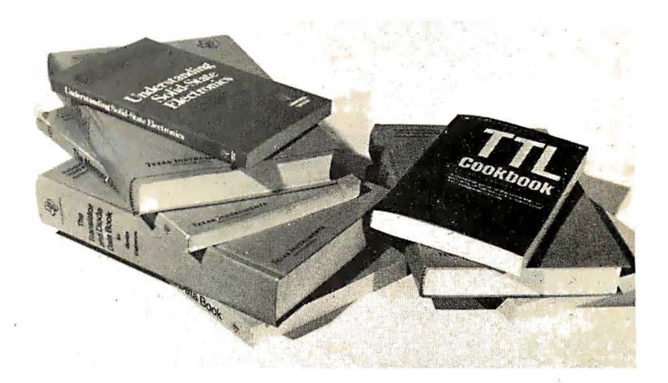
You may photocopy this page if you wish to leave your BYTE intact.

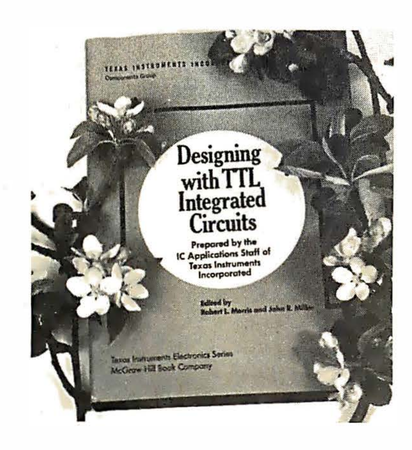
Prices shown are subject to change without notice.
All orders must be prepaid.
In unusual cases, processing may exceed 30 days.

Circle 12 on inquiry card.

A BIT More

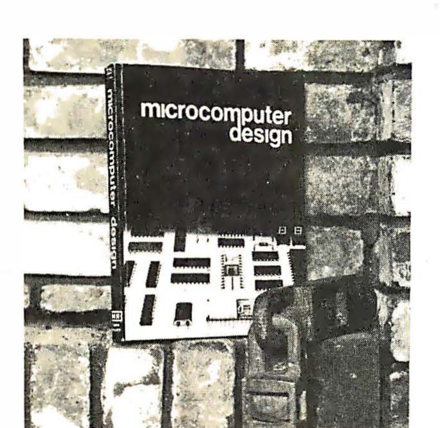
When you build a project, you need information. All you find in the advertisements for parts are mysterious numbers identifying the little beasties . . . hardly the sort of information which can be used to design a custom logic circuit. You can find out about many of the numbers by using the information found in these books. No laboratory bench is complete without an accompanying library shelf filled with references.





Order these absolutely essential references from Texas Instruments today:

___The TTL Data Book for Design Engineers, $4.95, new second edition.
___The Linear and Interface Circuits Data Book for Design Engineers, $3.95.
___The Semiconductor Memory Data Book for Design Engineers, $2.95.
——The Transistor and Diode Data Book for Design Engineers, $8.50.
——The Power Semiconductor Handbook for Design Engineers, $7.50.
___Understanding Solid State Electronics, $2.95.
___The Optoelectronics Data Book for Design Engineers, $2.95.
___Designing with TTL Integrated Circuits, edited by Robert L Morris and John R Miller, published by McGraw-Hill, $26.



——**The TTL Cookbook** by Don Lancaster, published by Howard W Sams. Start your quest for data here with Don's tutorial explanations of what makes a TTL logic design tick. 335 pages, $8.95.

——**Microcomputer Design** by Donald P Martin, Edited and published by Kerry S Berland, Martin Research. Purchase your copy of the definitive source for circuitry and hardware design information on the 8008 and 8080 computers. $14.95.

——Texas Instruments has just come out with the updated version of the **TTL Data Book for Design Engineers,** a new 832 page book which includes information formerly in the previous edition and its supplement, as well as new circuits which have arrived on the scene since publication of the earlier versions of this book. The second edition of the **TTL Data Book for Design Engineers** is your most up to date and current source of information on the design specifications and characteristics of the Texas Instruments 7400 series of devices. In it you'll find a complete section of 7400 series pinout diagrams at the front, plus the usual detailed descriptive information on the more complicated circuits. Only $4.95.

Send to:

BITS, Inc
70 Main St
Peterborough NH 03458

master charge   VISA

Check payment method:
___My check is enclosed
___Bill my MC No. ______________________ Exp. date ________
___Bill my VISA No. ______________________ Exp. date ________

Name ______________________
Address ______________________
City ______________ State ______ Zip ______

Total for all books checked $ ________
Postage, 50 cents per book
for ________ books $ ________
Grand Total $ ________

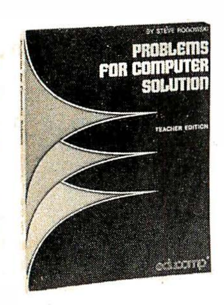
Signature ______________________

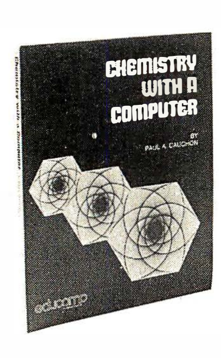
You may photocopy this page if you wish to leave your BYTE intact. Please allow six weeks for delivery.

Circle 12 on inquiry card.

Take a Byte of APL

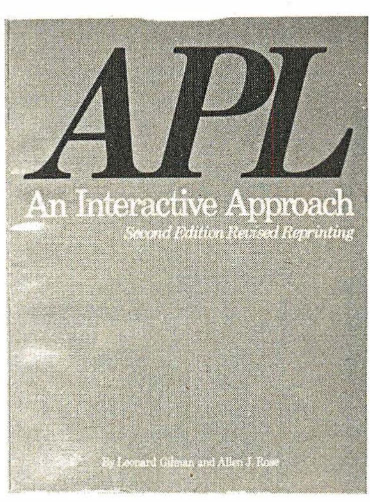
———**Problems for Computer Solution** by Steve Rogowski, Teacher Edition. A collection of mathematical problems designed to stimulate thought and encourage research by students towards the goal of a final solution. The subjects range from arithmetic through calculus and on to problems that have yet to be resolved. An analysis of each problem provides ideas on logical approaches to a solution, and a sample program demonstrates one possible final solution. $9.95.



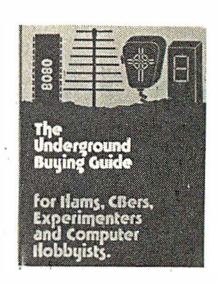


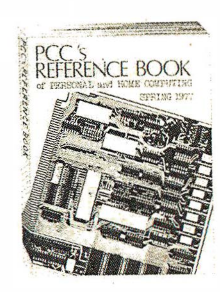
———**Chemistry with a Computer** by Paul A Cauchon. People are always looking for details of applications for the computer. It is one thing to say "gee whiz wouldn't it be nice if. . ." but such thoughts are but the stimulus to action. One result of a "gee whiz" idea is a series of educational and tutorial BASIC programs for use by teachers of chemistry, invented by Paul A Cauchon and published by Educomp Corporation. This book contains a collection of tutorial, simulation and problem generation programs which can be employed to advantage by teachers in high schools or colleges wherever a BASIC facility is available. If you're a chemistry professor or teacher by trade, or just a hobbyist interested in chemistry, this book will prove to be an invaluable tutorial aid. $9.95.

———**APL—An Interactive Approach** Second Edition, Revised, by Gilman and Rose. Here's an excellent way to introduce yourself to the APL language. APL is rapidly becoming one of the most popular high level languages in the computer field because of its clarity and conciseness. Gilman and Rose have extensively updated their popular book to include the latest information about the language and the various forms of it which are now in use. Since the examples are all carefully spelled out, **APL—An Interactive Approach** is particularly recommended for those who do not have access to an APL terminal. Answers to all problems are included. $11.95.



———**The Underground Buying Guide**, by Dennis A King. Here at last is a source book for all those hard to find suppliers! It's designed especially for computer hobbyists, experimenters, hams and CB'ers, and can tell you where to buy items like connectors, discrete components, electronic music supplies, instrumentation, analog to digital and digital to analog converters, and synthesizers. The list goes on to include Teletypes, speakers, microcomputer software, cassette units, floppy disks and many other items. It will be an invaluable addition to your reference library. $5.95.





———**PCC'S Reference Book of Personal and Home Computing.** Ever try to find the addresses of some manufacturers of, say, tape cassette or floppy disk interfaces for micros? Frustrating, isn't it? Well PCC has done something about it. This book lists hundreds of companies and stores selling hardware, software, and services. Survey articles on software, hardware, kits, applications and the future for the experienced and the not-so-experienced user of micros. Also included in this edition are bibliographies for further reference, book reviews, and an index of the articles from the major hobbyist magazines. $5.95

Send to:
**Bits, Inc.
70 Main Street
Peterborough NH 03458**

master charge — BANKAMERICARD

Check Payment method:
____My check is enclosed
____Bill my MC No.________________ Exp. date ________
____Bill my BAC No.________________ Exp. date ________

Name ______________________________

Address ______________________________

City ____________ State ________ Zip Code ____________

Signature ______________________________

Total for all books checked $________

Postage, 50 cents per book for______ books $________

Grand Total $________

Prices shown are subject to change without notice.
All orders must be prepaid.
In unusual cases, processing may exceed 30 days.

You may photocopy this page if you wish to leave your BYTE intact.

Circle 12 on inquiry card.

Continued from page 32

stations. Most receivers track a master and two slave stations. This produces two lines of position on a hyberbolic grid; the intersection is the position of the receiver. LORAN-C is a fairly accurate system giving fixes good to from 300 feet to ¼ mile, but it is limited in range. The range is up to 1200 to 1500 miles. Accuracy is degraded at night due to skywave interference. At present LORAN-C coverage is far from worldwide. Since it is principally a system for ship navigation, it is useful mainly in coastal areas. There are lots of receivers on the market, none of which is really inexpensive.

I am familiar with one satellite navigation system in current usage (others are in the developmental stage). This is the Navy Navigational Satellite System. It is a high accuracy system (100 feet for stationary receivers) which measures Doppler frequency of a 150 MHz and a 400 MHz carrier. The use of two frequencies allows for correction for ionospheric refraction. The satellite also transmits digital data which includes time of day and a description of the satellite's orbit. This is usually fed from a receiver into a minicomputer which processes the data and produces a fix in terms of latitude and longitude. This system does not produce continuous navigation since fixes can only be obtained when a satellite passes over the receiver. Since the satellites are in polar orbits the time between fixes normally ranges from 15 minutes to 4 hours with more frequent fixes as the receiver nears either pole. Again a number of companies manufacture receivers, but again they are expensive. A cheap SATNAV receiver and a set of microcomputer programs to produce a fix would certainly be an interesting, though formidable, project.

J Dean Clamons
Systems Analyst
Shipboard Computing Group
Naval Research Laboratory
Washington DC 20375



SHIMMY?

What causes the shimmy on my video monitor? Sometimes it looks like it's doing a hula.

John C Ford
9724 Tweedy Ln
Downy CA 90240

The problem superficially sounds like a beat between two frequencies. If the shimmy of which you speak is an amplitude modulation on an otherwise solid picture, you may have some form of cross talk with 60 Hz and some submultiple of your video interface's internal timing references. ■

Implementing an LSI Frequency Counter

Perry Lynne
990 Arques Av
Sunnyvale CA 94086

The new generation of programmable large scale integration (LSI) IO devices is proving to be as exciting as the microprocessors to which they are connected. With the aid of these LSI devices, complete functions can be added to a microprocessor system with only a few integrated circuits. One example of an LSI device with this kind of capability is the 8253 programmable interval timer which can be easily interfaced to almost any microprocessor. Using this device, a complete frequency counter can be constructed with just a couple of integrated circuits.

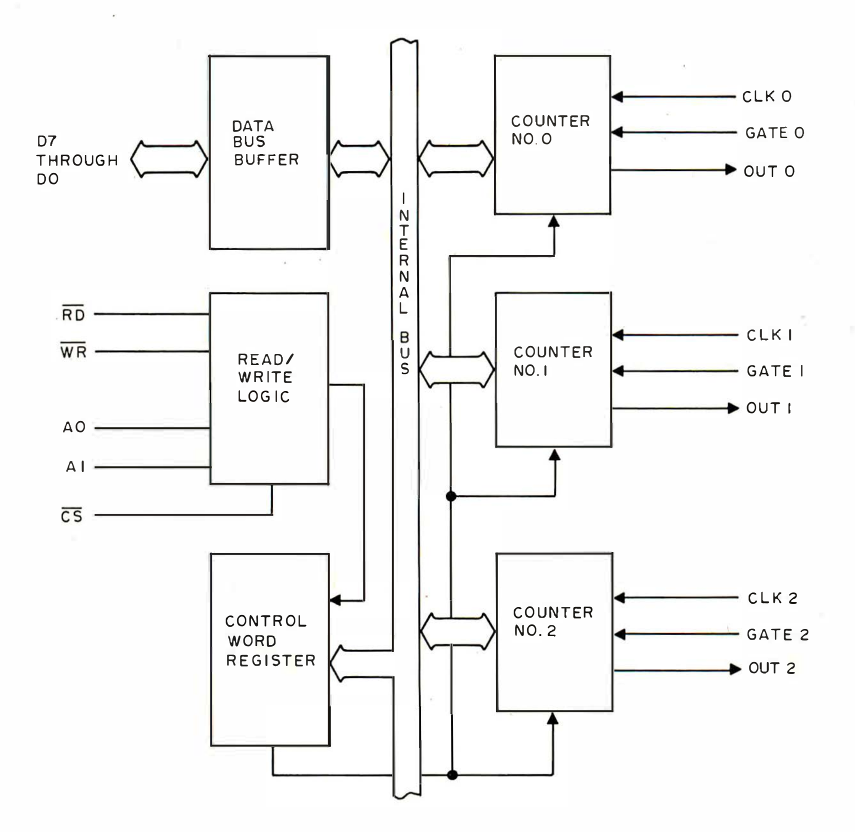


Figure 1: A block diagram of the 8253 programmable interval timer, which is made up of three independent down counters serviced by an internal bus. Software commands from the external processor are used to put the circuit in any one of the four different "modes" of operation. The modes, which are discussed in the mode definition box, include a mode which causes an interrupt on terminal count, a programmable oneshot, a rate generator and a square wave rate generator.

What's on This Chip?

The 8253 contains three independent 16 bit down counters (see figure 1). Each counter has a separate count input, gate input for gating the count and count output. Each can count in binary or binary coded decimal (BCD). Also, each counter can operate in one of four separate modes determined by storing a "mode" word in the device for each of the counters, usually on power up initialization. These mode words stay stored in the 8253 until they are changed by the microprocessor under software control.

In table 2 the format for loading the mode word and reading or loading the count in each counter is shown. The configuration of the mode word is shown in table 3.

The mode word for each counter on the 8253 determines whether the upper or lower half of the counter will be read or written, or whether the counter expects two sequential reads or writes to move 16 bits of data in or out of the device. The type of output the counters will produce is also determined by the mode word. See the shaded box on mode definitions for a discussion of each mode.

Programming the 8253

Each counter is individually programmed by writing a control word to location A1, A0 = 11. The control word format is as follows:

D7	D6	D5	D4
SC1	SC0	RL1	RL0
D3	D2	D1	D0
0	M1	M0	BCD

Where D0 - D7 is the contents of the data bus when the mode word is written to the 8253.

In this application the 8253 will be used as a single chip frequency counter. Figure 3 shows the functions of each counter. To determine the frequency of an unknown

The Two Modes of Operation of the 8253 Used in the Programmable Frequency Counter

Mode 1: Programmable Oneshot

The output will go low on the count following the rising edge of the gate/trigger input.

The output will go high on the terminal count. If a new count value is loaded while the output is low it will not affect the duration of the oneshot pulse until the succeeding trigger. The current count can be read at any time without affecting the oneshot pulse.

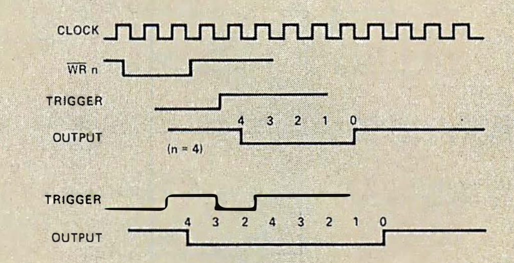
Mode 2: Rate Generator

Divide by N counter. The output will be low for one period of the input clock. The period from one output pulse to the next equals the number of input counts in the count register. If the count register is reloaded between output pulses the present period will not be affected, but the subsequent period will reflect the new value.

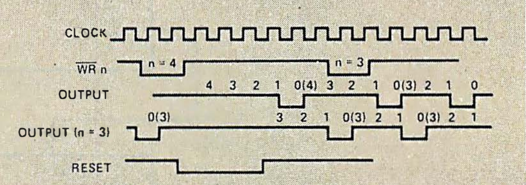
The gate/reset input, when low, will force the output high. When the gate/reset input goes high, the counter will start from the initial count. Thus, the gate/reset input can be used to synchronize the counter.

When this mode is set, the output will remain high until after the count register is loaded. The output then can also be synchronized by software.

MODE 1



MODE 2



Figures 2a and 2b: Two possible modes of operation for the 8253 programmable interval timer. (There are six in all.)

Signal Status / Modes	Low Or Going Low	Rising	High
1	— —	1) Initiates counting 2) Resets output after next clock	— —
2	1) Disables counting 2) Sets output immediately high	Initiates counting	Enables counting

Table 1: Gate pin operations summary.

Table 2: The configuration of the mode word, which determines whether the upper or lower half of the counter will be read or written, or whether the counter expects two sequential reads or writes to move 16 bits of data in or out of the device.

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A_1	A_0	
0	1	0	0	0	Load counter 0
0	1	0	0	1	Load counter 1
0	1	0	1	0	Load counter 2
0	1	0	1	1	Write mode word
0	0	1	0	0	Read counter 0
0	0	1	0	1	Read counter 1
0	0	1	1	0	Read counter 2
0	0	1	1	1	No-operation 3-state
1	X	X	X	X	Disable 3-state
0	1	1	X	X	No-operation 3-state

Where

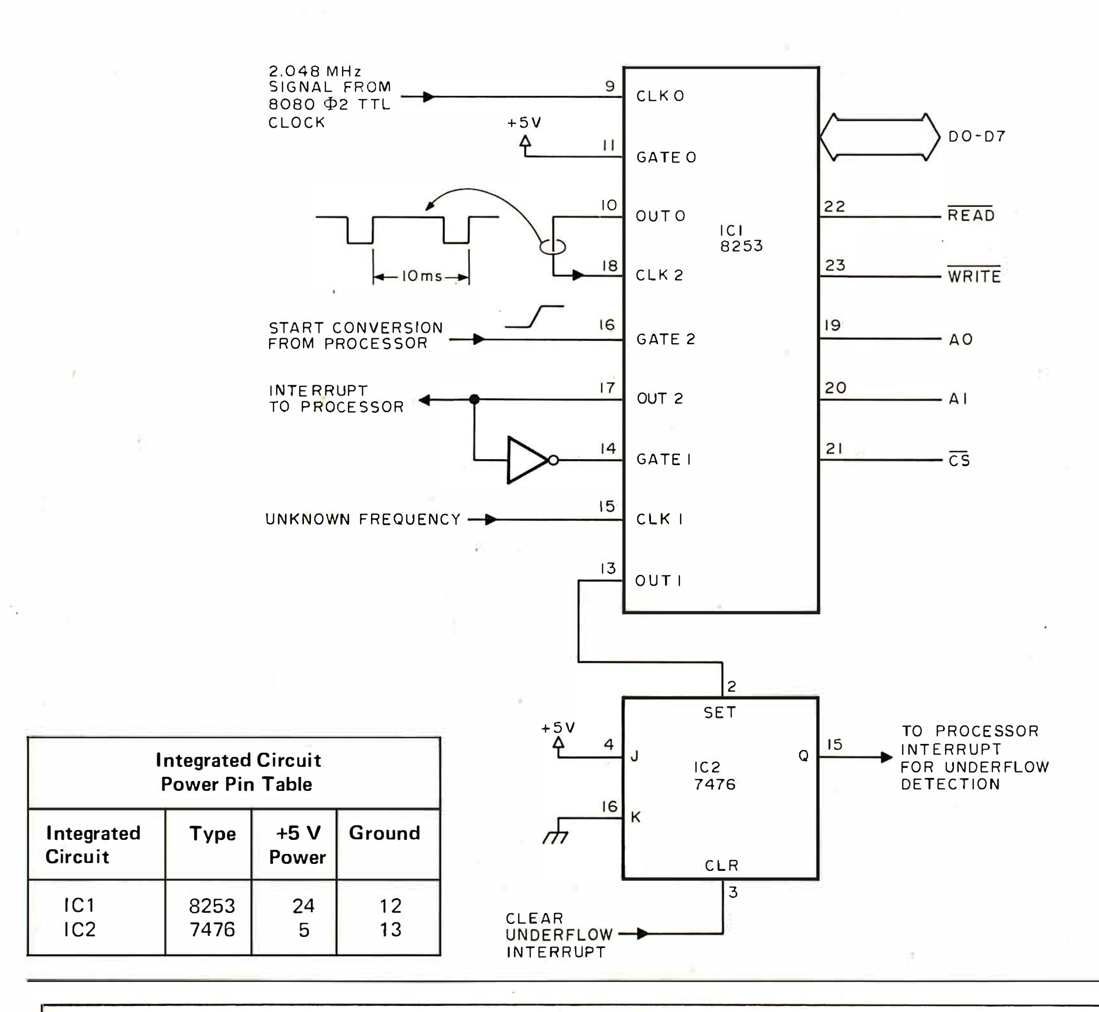
$\overline{CS}$ = Chip select
$\overline{RD}$ = Control signal to read data from the 8253
$\overline{WR}$ = Control signal to write data to the 8253
A_1A_0 = Address lines to select various sections of 8253

Table 3: The format for loading the mode word and reading or loading the count in each counter:

SC1	SC0	
0	0	Select counter 0
0	1	Select counter 1
1	0	Select counter 2
RL1	**RL0**	
1	0	Read/Load most significant byte only
0	1	Read/Load least significant byte only
1	1	Read/Load least significant byte first, then most significant byte.
M1	**M0**	**(M=Mode)**
0	0	Set Mode 0
0	1	Set Mode 1
1	0	Set Mode 2
1	1	Set Mode 3

BCD **(BCD=Binary Coded Decimal)**

0 Binary counter (16 bits)
1 BCD counter (4 decades)



Integrated Circuit Power Pin Table

Integrated Circuit	Type	+5 V Power	Ground
IC1	8253	24	12
IC2	7476	5	13

Figure 3: A two chip frequency counter. An unknown frequency is counted by counter number 1 during a precise time interval. This precise time interval is generated by counters 0 and 2. Counter 0 is programmed by the software to count the 8080 TTL clock input and divide it by 20,480. This creates a series of output pulses at a rate of one every 10 ms. These pulses are counted by counter 2 to produce a oneshot of programmable length. The oneshot is used to enable counter 1, which then counts the unknown frequency. Finally, the processor reads out the value of counter 1 and calculates the unknown frequency. IC2 is used to signal an underflow. This allows the circuit to count up to values beyond the normal maximum of 65,535 (for a 16 bit counter).

incoming signal we must count the number of cycles of the signal during a precise predetermined interval called the timebase.

In figure 3 we generate this timebase using counters 2 and 0. The 8080 Φ_2 TTL clock, which is crystal controlled at 2.048 MHz, is input to counter 0 operating mode 2. The output of this counter, which divides the count by 20,480, is a precise 10 ms signal. The output of counter 0 is then input to counter 2 which is operating in mode 1. In this mode counter 2 counts the 10 ms pulses and produces a oneshot output which serves as a precise interval for counting the unknown frequency.

This timebase interval is programmable by the microprocessor and can be varied from 100 ms to 100 seconds by storing the appropriate divider ratio in counter 2. This oneshot action is initiated under software control by strobing the gate input of counter 2. Note that after the strobe the oneshot action does not start until the next falling clock edge, so the interval is precise.

Next the output of counter 2 goes to the gate input of counter 1 which is operating in mode 2. This counter is allowed to count the unknown frequency of the incoming signal during the period that the gate input is high. The rising edge of the oneshot output of counter 2 is used to interrupt the microprocessor and signal that the frequency to digital conversion is complete. The processor then reads the resultant count in counter 1.

The software for servicing this progammable frequency counter is shown in listing 1. Note that, since these are down counters, the software initializes counter 1 with all 1s and the value stored in this counter at any particular time is the complement of the number of counts received from the unknown frequency. For example, if one count has been received, counter 1 will contain 1111 1111 1111 1110; the complement is 0000 0000 0000 0001.

The maximum count with a 16 bit counter is 65,535. (Note: with time base constants in listing set up for a 1 second count period, this 16 bit range measures frequencies from 1 Hz to 65,535 Hz). If the rising edge of the output of counter 1 (which signals an underflow) is used to interrupt the microprocessor, then the processor can count the number of interrupts in software. The processor can therefore keep a running total of the number of times the counter has passed through 65,535 counts and can therefore adjust the final count appropriately. This will enable counts much larger than 65,535 to be accumulated without having to use additional integrated circuits.■

Listing 1: Software for servicing the programmable frequency counter.

```
; INITIALIZING THE 8253 COUNTERS FOR
; THEIR VARIOUS MODES THE 8253 IS
; CONNECTED IN A MEMORY MAPPED IO
; CONFIGURATION IN THIS APPLICATION AND
; THEREFORE IS ADDRESSED THROUGH MEMORY
; REFERENCE INSTRUCTIONS

        LXI    H,P8253        ; INITIALIZE MEMORY POINTER
                              ; TO 8253 MODE WORD
        MVI    M,COUNT 0      ; INITIALIZE COUNTER 0 TO
                              ; MODE 2
        MVI    M,COUNT 1      ; INITIALIZE COUNTER 1 TO MODE 2
        MVI    M,COUNT 2      ; INITIALIZE COUNTER 2 TO MODE 1

; INITIALIZE COUNTER 2 WITH DIVIDER RATIO
; TO PRODUCE APPROPRIATE TIMEBASE

        DCX    H              ; POINT TO COUNTER 2
        LXI    B,TIMEBASE     ; TIMEBASE = 3E8H FOR 10SEC
        MOV    M,C            ; = 64H FOR 1SEC
        MOV    M,B            ; = 0AH FOR 100MS

; INITIALIZE COUNTER 1 WITH ALL 1s SINCE
; THIS COUNTER WILL BE COUNTING DOWN
        DCX    H              ; POINT TO COUNTER 1
        MVI    M,OFFH
        MVI    M,OFFH

; INITIALIZE COUNTER 0 WITH A DIVIDE BY 20480.
        DCX    H              ; POINT TO COUNTER 0
        MVI    M,00H
        MVI    M,50H

; THIS SUBROUTINE SERVICES THE FREQUENCY
; COUNTER INTERRUPT BY READING THE FREQUENCY IN
; COUNTER 0 AND STARTING A NEW CYCLE

COUNTDONE: PUSH   A              ; SAVE REGISTERS WHICH ARE
           PUSH   H              ; MODIFIED BY THIS ROUTINE
           LXI    H,P8253-2      ; POINT TO COUNTER 1
           MOV    A,M            ; GET LSB OF RESULT
           CMA                   ; COMPLEMENT THE DATA
           STA    COUNTRSLT      ; STORE IN COUNT RESULT
           MOV    A,M            ; GET MSB OF RESULT
           CMA                   ; COMPLEMENT DATA
           STA    COUNTRSLT+1    ; STORE AWAY
           MVI    M,OFFH         ; STORE ALL 1s IN
           MVI    M,OFFH         ; COUNTER 1
           OUT    START          ; CLEAR INTERRUPT AND
                                 ; START NEW CYCLE
           POP    H              ; RESTORE STATUS
           POP    A              ; AND RETURN
           EI
           RET

; THIS ROUTINE SERVICES AN OVERFLOW
; INTERRUPT FROM COUNTER 1 AND
; KEEPS A RUNNING TOTAL OF
; THE NUMBER OF OVERFLOWS
; FOR THIS CYCLE. OTHER SOFTWARE
; IN THE SYSTEM SHOULD CLEAR
; THIS QUANTITY WHEN A NEW CYCLE
; IS STARTED.

OVERFLOW:

        PUSH   A              ; SAVE SYSTEM STATUS
        PUSH   H
        LXI    H,OVFLO        ; INCREMENT OVFLO
        INC    M
        OUT    CLINT          ; CLEAR THE INTERRUPT
        POP    H
        POP    A
        EI
        RET
```

Stephen Wozniak
Apple Computer
20863 Stevens Creek Blvd, B3C
Cupertino CA 95014

SWEET16: The 6502 Dream Machine

While writing Apple BASIC for a 6502 microprocessor I repeatedly encountered a variant of Murphy's Law. Briefly stated, any routine operating on 16 bit data will require at least twice the code that it should. Programs making extensive use of 16 bit pointers (such as compilers, editors and assemblers) are included in this category. In my case, even the addition of a few double byte instructions to the 6502 would have only slightly alleviated the problem. What I really needed was a hybrid of the MOS Technology 6502 and RCA 1800 architectures, a powerful 8 bit data handler complemented by an easy to use processor with an abundance of 16 bit registers and excellent pointer capability. My solution was to implement a nonexistent 16 bit "metaprocessor" in software, interpreter style, which I call SWEET16. This metaprocessor was sketched at the end of my article in May 1977 BYTE, and the purpose of this article is to fill in the details of SWEET16.

SWEET16 is based around sixteen 16 bit registers called R0 to R15, actually implemented as 32 memory locations. R0 doubles as the SWEET16 accumulator (ACC), R15 as the program counter (PC), and R14 as the status register. R13 holds compare instruction results and R12 is the subroutine return stack pointer if SWEET16 subroutines are used. All other SWEET16 registers are at the user's unrestricted disposal.

SWEET16 instructions fall into register and nonregister categories. The register operations specify one of the 16 registers to be used as either a data element or a pointer to data in memory depending on the specific instruction. For example, the instruction INR R5 uses R5 as data and ST @R7 uses R7 as a pointer to data in memory. Except for the SET instruction, register operations only require one byte. The nonregister operations are primarily 6502 style branches with the second byte specifying a ±127 byte displacement relative to the address of the following instruction. If a prior register operation result meets a specified branch condition, the displacement is added to SWEET16's program counter, effecting a branch.

SWEET16 is intended as a 6502 enhancement package, not a stand alone processor. A 6502 program switches to SWEET16 mode with a subroutine call, and subsequent code is interpreted as SWEET16 instructions. The nonregister operation RTN returns the user program to the 6502's direct execution mode after restoring the internal register contents (A, X, Y, P and S). The example of listing 1 illustrates how to use SWEET16 in some program segment.

```
             300  B9 00 02             LDA  IN,Y      Get a char.
             303  C9 CD                CMP  "M"       "M" for move?
             305  D0 09                BNE  NOMOVE      No, skip move.
             307  20 00 08             JSR  SW16        Yes, call SWEET16.
         (   30A  41         MLOOP     LD   @R1       R1 holds source address.
         |   30B  52                   ST   @R2       R2 holds dest. address.
SWEET16 <    30C  F3                   DCR  R3        Decrement length.
         |   30D  07 FB                BNZ  MLOOP       Loop until done.
         (   30F  00                   RTN            Return to 6502 mode.
             310  C9 C5      NOMOVE    CMP  "E"       "E" char?
             312  D0 13                BEQ  EXIT        Yes, exit.
             314  C8                   INY              No, continue.
```

Note: Registers A, X, Y, P and S are not disturbed by SWEET16.

Listing 1: Use of SWEET16 within an assembly language program is accomplished by executing a subroutine call to the SWEET16 entry point (address 307 here). This call preserves the processor registers at the time of entry and begins interpretive execution. End of interpretive execution is signaled by a RTN operation code of SWEET16, at which point all the processor registers will be restored.

Instruction Descriptions

The SWEET16 op code list is short and uncomplicated. Excepting relative branch displacements, hand assembly is trivial. All register op codes are formed by combining two hexadecimal digits, one for the op code and one to specify a register. For example,

op codes 15 and 45 both specify register R5 while codes 23, 27 and 29 are all ST (store) operations. Most register operations of SWEET16 are assigned to numerically adjacent pairs to facilitate remembering them. Thus LD and ST are op codes 2n and 3n respectively, while LD @ and ST @ are codes 4n and 5n.

Operation codes 00 to 0C (hexadecimal) are assigned to the 13 nonregister operations. Except for RTN (op code 0), BK (0A), and RS (B), the nonregister operations are 6502 style relative branches. The second byte of a branch instruction contains a ±127 byte displacement value (in two's complement form) relative to the address of the instruction immediately following the branch. If a specified branch condition is met by the prior register operation result, the displacement is added to the program counter effecting a branch. Except for BR (Branch always) and BS (Branch to Subroutine), the branch operation codes are assigned in complementary pairs, rendering them easily remembered for hand coding. For example, Branch if Plus and Branch if Minus are op codes 04 and 05, while Branch if Zero and Branch if NonZero are op codes 06 and 07.

Theory of Operation

SWEET16 execution mode begins with a subroutine call to SW16 (see listing 2, an assembly of SWEET16). The user must insure that the 6502 is in hexadecimal mode upon entry. [*For those unfamiliar with the 6502, arithmetic is either decimal or hexadecimal (binary) depending on a programmable flag. . .CH*] All 6502 registers are saved at this time, to be restored when a SWEET16 RTN instruction returns control to the 6502. If you can tolerate indefinite 6502 register contents upon exit, approximately 30 μs may be saved by entering SWEET16 at location SW16 + 3. Because this might cause an inadvertent switch from hexadecimal to decimal mode, it is advisable to enter at SW16 the first time through.

After saving the 6502 registers, SWEET16 initializes its program counter (R15) with the subroutine return address off the 6502 stack. SWEET16's program counter points to the location preceding the next instruction to be executed. Following the subroutine call are 1 byte, 2 byte, or 3 byte long SWEET16 instructions, stored in ascending

Listing 2: SWEET16 assembly. The SWEET16 program, assembled to reside at location 800 hexadecimal, is presented by this listing. The primary entry point is at the beginning, location SW16. An alternate entry point if there is no need to save processor registers is at location 803 in this assembly, SW16+3.

```
                          SWEET16 INTERPRETER
11:18 A.M., THU, MAY 12, 1977

                 ***********************
                 *                     *
                 *   APPLE-II PSEUDO   *
                 * MACHINE INTERPRETER *
                 *                     *
                 *      S. WOZNIAK     *
                 * APPLE COMPUTER INC  *
                 *                     *
                 ***********************
                  TITLE "SWEET16 INTERPRETER"
                 R0L       EPZ  $0
                 R0H       EPZ  $1
                 R14H      EPZ  $1D
                 R15L      EPZ  $1E
                 R15H      EPZ  $1F
                 S16PAG    EQU  $F7
                           ORG  $800
0800: 20 74 09   SW16      JSR  SAVE      PRESERVE 6502 REG CONTENTS
0803: 68                   PLA
0804: 85 1E                STA  R15L      INIT SWEET16 PC
0806: 68                   PLA            FROM RETURN
0807: 85 1F                STA  R15H        ADDRESS
0809: 20 0F 08   SW16B     JSR  SW16C     INTERPRET AND EXECUTE
080C: 4C 09 08             JMP  SW16B     ONE SWEET16 INSTR.
080F: E6 1E      SW16C     INC  R15L
0811: D0 02                BNE  SW16D     INCR SWEET16 PC FOR FETCH
0813: E6 1F                INC  R15H
0815: A9 F7      SW16D     LDA  #S16PAG
0817: 48                   PHA            PUSH ON STACK FOR RTS
0818: A0 00                LDY  #$0
081A: B1 1E                LDA  (R15L),Y FETCH INSTR
081C: 29 0F                AND  #$F       MASK REG SPECIFICATION
081E: 0A                   ASL  A         DOUBLE FOR 2-BYTE REG'S
081F: AA                   TAX            TO X-REG FOR INDEXING
0820: 4A                   LSR  A
0821: 51 1E                EOR  (R15L),Y NOW HAVE OPCODE
0823: F0 0B                BEQ  TOBR      IF ZERO THEN NON-REG OP
0825: 86 1D                STX  R14H      INDICATE'PRIOR RESULT REG'
0827: 4A                   LSR  A
0828: 4A                   LSR  A         OPCODE*2 TO LSB'S
0829: 4A                   LSR  A
082A: A8                   TAY            TO Y-REG FOR INDEXING
082B: B9 58 08             LDA  OPTBL-2,Y LOW-ORDER ADR BYTE
082E: 48                   PHA            ONTO STACK
082F: 60                   RTS            GOTO REG-OP ROUTINE
0830: E6 1E      TOBR      INC  R15L
0832: D0 02                BNE  TOBR2     INCR PC
0834: E6 1F                INC  R15H
0836: BD 5B 08   TOBR2     LDA  BRTBL,X   LOW-ORDER ADR BYTE
0839: 48                   PHA            ONTO STACK FOR NON-REG OP
083A: A5 1D                LDA  R14H      'PRIOR RESULT REG' INDEX
083C: 4A                   LSR  A         PREPARE CARRY FOR BC, BNC.
083D: 60                   RTS            GOTO NON-REG OP ROUTINE
083E: 68         RTNZ      PLA            POP RETURN ADDRESS
083F: 68                   PLA
0840: 20 7F 09             JSR  RESTORE   RESTORE 6502 REG CONTENTS
0843: 6C 1E 00             JMP  (R15L)    RETURN TO 6502 CODE VIA PC
0846: B1 1E      SETZ      LDA  (R15L),Y HIGH-ORDER BYTE OF CONST
0848: 95 01                STA  R0H,X
084A: 88                   DEY
084B: B1 1E                LDA  (R15L),Y LOW-ORDER BYTE OF CONSTANT
084D: 95 00                STA  R0L,X
084F: 98                   TYA            Y-REG CONTAINS 1
0850: 38                   SEC
0851: 65 1E                ADC  R15L      ADD 2 TO PC
0853: 85 1E                STA  R15L
0855: 90 02                BCC  SET2
0857: E6 1F                INC  R15H
0859: 60         SET2      RTS
085A: 79         OPTBL     DFB  SET-1     (1X)
085B: 70         BRTBL     DFB  RTN-1     (0)
085C: 7B                   DFB  LD-1      (2X)
085D: 14                   DFB  BR-1      (1)
085E: 84                   DFB  ST-1      (3X)
085F: 15                   DFB  BNC-1     (2)
0860: 9C                   DFB  LDAT-1    (4X)
0861: 26                   DFB  BC-1      (3)
0862: 8D                   DFB  STAT-1    (5X)
0863: 29                   DFB  BP-1      (4)
0864: BE                   DFB  LDDAT-1   (6X)
0865: 30                   DFB  BM-1      (5)
0866: C8                   DFB  STDAT-1   (7X)
0867: 37                   DFB  BZ-1      (6)
0868: A6                   DFB  POP-1     (8X)
0869: 40                   DFB  BNZ-1     (7)
086A: D2                   DFB  STPAT-1   (9X)
086B: 49                   DFB  BM1-1     (8)
086C: FC                   DFB  ADD-1     (AX)
086D: 54                   DFB  BNM1-1    (9)
086E: E5                   DFB  SUB-1     (BX)
086F: 7C                   DFB  BK-1      (A)
0870: AA                   DFB  POPD-1    (CX)
0871: 5F                   DFB  RS-1      (B)
0872: E7                   DFB  CPR-1     (DX)
0873: 0A                   DFB  BS-1      (C)
0874: 95                   DFB  INR-1     (EX)
0875: 5E                   DFB  NUL-1     (D)
```

Listing 2, continued:

```
Ø876: DC        ØØØ98           DFB  DCR-1      (FX)
Ø877: 5E        ØØØ99           DFB  NUL-1      (E)
Ø878: 5E        ØØ1ØØ           DFB  NUL-1      (UNUSED)
Ø879: 5E        ØØ1Ø1           DFB  NUL-1      (F)
Ø87A: 1Ø CA     ØØ1Ø2 SET       BPL  SETZ       ALWAYS TAKEN
Ø87C: B5 ØØ     ØØ1Ø3 LD        LDA  RØL,X
                ØØ1Ø4 BK        EQU  *-1
Ø87E: 85 ØØ     ØØ1Ø5           STA  RØL
Ø88Ø: B5 Ø1     ØØ1Ø6           LDA  RØH,X      MOVE RX TO RØ
Ø882: 85 Ø1     ØØ1Ø7           STA  RØH
Ø884: 6Ø        ØØ1Ø8           RTS
Ø885: A5 ØØ     ØØ1Ø9 ST        LDA  RØL
Ø887: 95 ØØ     ØØ11Ø           STA  RØL,X      MOVE RØ TO RX
Ø889: A5 Ø1     ØØ111           LDA  RØH
Ø88B: 95 Ø1     ØØ112           STA  RØH,X
Ø88D: 6Ø        ØØ113           RTS
Ø88E: A5 ØØ     ØØ114 STAT      LDA  RØL
Ø89Ø: 81 ØØ     ØØ115 STAT2     STA  (RØL,X)    STORE BYTE INDIRECT
Ø892: AØ ØØ     ØØ116           LDY  #$Ø
Ø894: 84 1D     ØØ117 STAT3     STY  R14H       INDICATE RØ IS RESULT REG
Ø896: F6 ØØ     ØØ118 INR       INC  RØL,X
Ø898: DØ Ø2     ØØ119           BNE  INR2       INCR RX
Ø89A: F6 Ø1     ØØ12Ø           INC  RØH,X
Ø89C: 6Ø        ØØ121 INR2      RTS
Ø89D: A1 ØØ     ØØ122 LDAT      LDA  (RØL,X)    LOAD INDIRECT (RX)
Ø89F: 85 ØØ     ØØ123           STA  RØL        TO RØ
Ø8A1: AØ ØØ     ØØ124           LDY  #$Ø
Ø8A3: 84 Ø1     ØØ125           STY  RØH        ZERO HIGH-ORDER RØ BYTE
Ø8A5: FØ ED     ØØ126           BEQ  STAT3      ALWAYS TAKEN
Ø8A7: AØ ØØ     ØØ127 POP       LDY  #$Ø        HIGH ORDER BYTE = Ø
Ø8A9: FØ Ø6     ØØ128           BEQ  POP2       ALWAYS TAKEN
Ø8AB: 2Ø DD Ø8  ØØ129 POPD      JSR  DCR        DECR RX
Ø8AE: A1 ØØ     ØØ13Ø           LDA  (RØL,X)    POP HIGH-ORDER BYTE @RX
Ø8BØ: A8        ØØ131           TAY             SAVE IN Y-REG
Ø8B1: 2Ø DD Ø8  ØØ132 POP2      JSR  DCR        DECR RX
Ø8B4: A1 ØØ     ØØ133           LDA  (RØL,X)    LOW-ORDER BYTE
Ø8B6: 85 ØØ     ØØ134           STA  RØL        TO RØ
Ø8B8: 84 Ø1     ØØ135           STY  RØH
Ø8BA: AØ ØØ     ØØ136 POP3      LDY  #$Ø        INDICATE RØ AS LAST
Ø8BC: 84 1D     ØØ137           STY  R14H            RESULT REG
Ø8BE: 6Ø        ØØ138           RTS
Ø8BF: 2Ø 9D Ø8  ØØ139 LDDAT     JSR  LDAT       LOW BYTE TO RØ, INCR RX
Ø8C2: A1 ØØ     ØØ14Ø           LDA  (RØL,X)    HIGH-ORDER BYTE TO RØ
Ø8C4: 85 Ø1     ØØ141           STA  RØH
Ø8C6: 4C 96 Ø8  ØØ142           JMP  INR        INCR RX
Ø8C9: 2Ø 8E Ø8  ØØ143 STDAT     JSR  STAT       STORE INDIRECT LOW-ORDER
Ø8CC: A5 Ø1     ØØ144           LDA  RØH        BYTE AND INCR RX.  THEN
Ø8CE: 81 ØØ     ØØ145           STA  (RØL,X)    STORE HIGH-ORDER BYTE.
Ø8DØ: 4C 96 Ø8  ØØ146           JMP  INR        INCR RX AND RETURN
Ø8D3: 2Ø DD Ø8  ØØ147 STPAT     JSR  DCR        DECR RX
Ø8D6: A5 ØØ     ØØ148           LDA  RØL
Ø8D8: 81 ØØ     ØØ149           STA  (RØL,X)    STORE RØ LOW BYTE @RX
Ø8DA: 4C BA Ø8  ØØ15Ø           JMP  POP3       INDICATE RØ AS LAST
Ø8DD: B5 ØØ     ØØ151 DCR       LDA  RØL,X           RESULT REG
Ø8DF: DØ Ø2     ØØ152           BNE  DCR2       DECR RX
Ø8E1: D6 Ø1     ØØ153           DEC  RØH,X
Ø8E3: D6 ØØ     ØØ154 DCR2      DEC  RØL,X
Ø8E5: 6Ø        ØØ155           RTS
Ø8E6: AØ ØØ     ØØ156 SUB       LDY  #$Ø        RESULT TO RØ
Ø8E8: 38        ØØ157 CPR       SEC             NOTE Y-REG = 13*2 FOR CPR
Ø8E9: A5 ØØ     ØØ158           LDA  RØL
Ø8EB: F5 ØØ     ØØ159           SBC  RØL,X
Ø8ED: 99 ØØ ØØ  ØØ16Ø           STA  RØL,Y      RØ-RX TO RY
Ø8FØ: A5 Ø1     ØØ161           LDA  RØH
Ø8F2: F5 Ø1     ØØ162           SBC  RØH,X
Ø8F4: 99 Ø1 ØØ  ØØ163 SUB2      STA  RØH,Y
Ø8F7: 98        ØØ164           TYA             LAST RESULT REG*2
Ø8F8: 69 ØØ     ØØ165           ADC  #$Ø        CARRY TO LSB
Ø8FA: 85 1D     ØØ166           STA  R14H
Ø8FC: 6Ø        ØØ167           RTS
Ø8FD: A5 ØØ     ØØ168 ADD       LDA  RØL
Ø8FF: 75 ØØ     ØØ169           ADC  RØL,X
Ø9Ø1: 85 ØØ     ØØ17Ø           STA  RØL        RØ+RX TO RØ
Ø9Ø3: A5 Ø1     ØØ171           LDA  RØH
Ø9Ø5: 75 Ø1     ØØ172           ADC  RØH,X
Ø9Ø7: AØ ØØ     ØØ173           LDY  #$Ø        RØ FOR RESULT
Ø9Ø9: FØ E9     ØØ174           BEQ  SUB2       FINISH ADD
Ø9ØB: A5 1E     ØØ175 BS        LDA  R15L       NOTE X-REG IS 12*2!
Ø9ØD: 2Ø 9Ø Ø8  ØØ176           JSR  STAT2      PUSH LOW PC BYTE VIA R12
Ø91Ø: A5 1F     ØØ177           LDA  R15H
Ø912: 2Ø 9Ø Ø8  ØØ178           JSR  STAT2      PUSH HIGH-ORDER PC BYTE
Ø915: 18        ØØ179 BR        CLC
Ø916: BØ ØE     ØØ18Ø BNC       BCS  BNC2       NO CARRY TEST
Ø918: B1 1E     ØØ181 BR1       LDA  (R15L),Y DISPLACEMENT BYTE
Ø91A: 1Ø Ø1     ØØ182           BPL  BR2
Ø91C: 88        ØØ183           DEY
Ø91D: 65 1E     ØØ184 BR2       ADC  R15L       ADD TO PC
Ø91F: 85 1E     ØØ185           STA  R15L
Ø921: 98        ØØ186           TYA
Ø922: 65 1F     ØØ187           ADC  R15H
Ø924: 85 1F     ØØ188           STA  R15H
Ø926: 6Ø        ØØ189 BNC2      RTS
Ø927: BØ EC     ØØ19Ø BC        BCS  BR
Ø929: 6Ø        ØØ191           RTS
Ø92A: ØA        ØØ192 BP        ASL  A          DOUBLE RESULT-REG INDEX
Ø92B: AA        ØØ193           TAX             TO X-REG FOR INDEXING
Ø92C: B5 Ø1     ØØ194           LDA  RØH,X      TEST FOR PLUS
Ø92E: 1Ø E8     ØØ195           BPL  BR1        BRANCH IF SO
Ø93Ø: 6Ø        ØØ196           RTS
Ø931: ØA        ØØ197 BM        ASL  A          DOUBLE RESULT-REG INDEX
Ø932: AA        ØØ198           TAX
Ø933: B5 Ø1     ØØ199           LDA  RØH,X      TEST FOR MINUS
Ø935: 3Ø E1     ØØ2ØØ           BMI  BR1
Ø937: 6Ø        ØØ2Ø1           RTS
Ø938: ØA        ØØ2Ø2 BZ        ASL  A          DOUBLE RESULT-REG INDEX
Ø939: AA        ØØ2Ø3           TAX
Ø93A: B5 ØØ     ØØ2Ø4           LDA  RØL,X      TEST FOR ZERO
Ø93C: 15 Ø1     ØØ2Ø5           ORA  RØH,X      (BOTH BYTES)
Ø93E: FØ D8     ØØ2Ø6           BEQ  BR1        BRANCH IF SO
Ø94Ø: 6Ø        ØØ2Ø7           RTS
Ø941: ØA        ØØ2Ø8 BNZ       ASL  A          DOUBLE RESULT-REG INDEX
Ø942: AA        ØØ2Ø9           TAX
Ø943: B5 ØØ     ØØ21Ø           LDA  RØL,X      TEST FOR NONZERO
Ø945: 15 Ø1     ØØ211           ORA  RØH,X      (BOTH BYTES)
Ø947: DØ CF     ØØ212           BNE  BR1        BRANCH IF SO
Ø949: 6Ø        ØØ213           RTS
Ø94A: ØA        ØØ214 BM1       ASL  A          DOUBLE RESULT-REG INDEX
Ø94B: AA        ØØ215           TAX
Ø94C: B5 ØØ     ØØ216           LDA  RØL,X      CHECK BOTH BYTES
Ø94E: 35 Ø1     ØØ217           AND  RØH,X      FOR $FF (MINUS 1)
```

memory locations like 6502 instructions. The main loop at SW16B repeatedly calls the "execute instruction" routine at SW16C which examines one op code for type and branches to the appropriate subroutine to execute it.

Subroutine SW16C increments the program counter (R15) and fetches the next op code which is either a register operation of the form OP REG (2 hexadecimal digits) with OP between hexadecimal 1 and F, or a nonregister operation of the form 0 OP with OP between hexadecimal 0 and D. Assuming a register operation, the register specification is doubled to account for the 2 byte SWEET16 registers and placed in the X register for indexing. Then the instruction type is determined. Register operations place the doubled register specification in the high order byte of R14 indicating the "prior result register" to subsequent branch instructions. Nonregister operations treat the register specification (right-hand half-byte) as their op code, increment the SWEET16 PC to point at the displacement byte of branch instructions, load the A-Reg with the "prior result register" index for branch condition testing, and clear the Y-Reg.

When Is an RTS Really a JSR?

Each instruction type has a corresponding subroutine. The subroutine entry points are stored in a table which is directly indexed by the op code. By assigning all the entries to a common page, only a single byte of address need be stored per routine. The 6502 indirect jump might have been used as follows to transfer control to the appropriate subroutine:

```
LDA #ADRH    High order address byte
STA IND+1
LDA OPTBL,X  Low order byte
STA IND
JMP (IND)
```

To save code the subroutine entry address (minus 1) is pushed onto the stack, high order byte first. A 6502 RTS (ReTurn from Subroutine) is used to pop the address off the stack and into the 6502 program counter (after incrementing by 1). The net result is that the desired subroutine is reached by executing a subroutine return instruction! [*This ironic situation is an example of what is commonly referred to as "cleverness."*]

Op Code Subroutines

The register operation routines make use of the 6502 "zero page indexed by X" and "indexed by X indirect" addressing modes to access the specified registers and indirect data. The "result" of most register ops is left

The Art of Computer Programming

Praised by many critics as the best books in their field, **The Art of Computer Programming, Volumes I, II and III,** are part of a projected seven volume omnibus survey of computer science now being completed by Donald E Knuth.

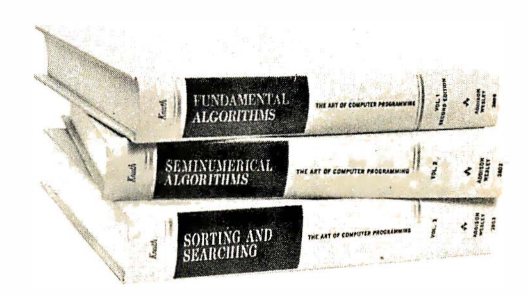
———Volume I, **Fundamental Algorithms,** begins with a thorough discussion of the mathematics used in computer programming, followed by a treatment of information structures, stacks, arrays, linked lists, dynamic storage allocation, and trees. 634 pp; $20.95.

———Volume II, **Seminumerical Algorithms,** is concerned with random numbers, statistical tests, random sequences, as well as arithmetic (floating point and multiple precision), polynomials, and rational arithmetic. 624 pp; $20.95.

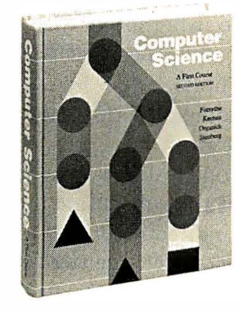
———Volume III deals with **Searching and Sorting,** and as the name implies, the emphasis is on algorithms for sorting, including combinatorial properties of permutations, internal sorting, optimum sorting, and external sorting. Also included is a section on sequential searching, hashing, digital searching, and more. 722 pp; $20.95.

A hypothetical assembly language called MIX has been developed by the author to illustrate programming examples throughout the series. MIX is easily convertible to other assembly languages.

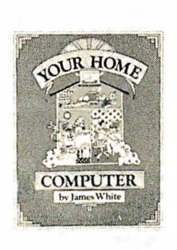
Prof Knuth writes with style and wit (among many memorable quotes is one from *McCall's Cookbook!*). This classic work belongs on the reference shelf of everyone seriously interested in computer science.

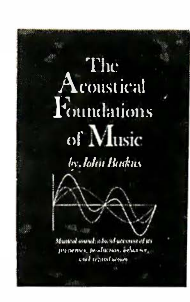


.... and Some of the Best for Beginners









———**Computer Science — A First Course.** The title of this book belies its thoroughness and rigor. **Computer Science: A First Course** is one of the most complete and well laid out computer science books we've seen in quite a while. This 760+ page book is for the serious experimenter who wants a definitive treatment of every aspect of software design and development. Chapters deal with algorithms, flowcharts, looping structures, stepwise decomposition, trees, storage concepts, interpreting and compiling, data processing, numerical processing, character strings, and a discussion of SAMOS, a special hypothetical computer system used throughout the book to illustrate computer science principles. **Computer Science: A First Course** makes an excellent source book. $16.95, Hardcover.

———**Understanding Microcomputers and Small Computer Systems** by Nat Wadsworth. What does "hexadecimal" mean? What is Boolean Logic? What's the difference between a PROM and an EROM? These questions and more are answered in Nat Wadsworth's **Understanding Microcomputers and Small Computer Systems.** You'll learn about the basic operation of a microcomputer, the types of instructions that microcomputers are able to carry out, machine language and higher level language programming, input and output devices, etc. The book concludes with a discussion of system considerations. The combined index and glossary is a useful bonus. If you're a beginner in the personal computing field, get a copy today! This 300 page book is only $9.95.

———**The Acoustical Foundations of Music,** subtitled **Musical Sound: a lucid account of its properties, production, behavior, and reproduction,** by John Backus. One of the problems involved in trying to produce music by computer is the scarcity of useful reference material on the basic theory of sound generation and musical acoustics. John Backus has filled this gap **with The Acoustical Foundations of Music,** a readable and informative guide that covers the physiological properties of sound, the ear and its perception of sounds, the effect of acoustic environment, the acoustical behavior of musical instruments, and the various applications of electronics and computers to the production, reproduction and composition of music.

The book assumes some slight knowledge of music on the part of the reader, but covers enough elementary physics and acoustics to orient the reader who has no knowledge of physics.

This clear and lively exposition of music's physical basis will be invaluable to all those experimenters interested in creating music by computer. 312 pages, and only $9.95 in hardcover.

———**Your Home Computer,** by James White, is a clearly written nontechnical description of personal computers that requires no prior knowledge of computers or electronics. The emphasis is on understanding; over 100 illustrations are included. Topics include: computing and you; communication inside a computer; computer thought processes; fixed memory; inputs and outputs; peripherals; systems components; how to choose a microcomputer; and so on. **Your Personal Computer** is the ideal book for readers who thought they could never understand how computers work. And the best part is that it's easy and fun to read. Yours for $6.

Send to:
BITS, Inc
70 Main Street
Peterborough NH 03458

Check Payment method:
____My check is enclosed
____Bill my MC No.______________ Exp. date ______
____Bill my BAC No.______________ Exp. date ______

Name ____________________

Address ____________________

City ________ State ______ Zip Code ________

Signature ____________________

Total for all books checked $______

Postage, 50 cents per book for ____ books $______

Grand Total $______

You may photocopy this page if you wish to leave your BYTE intact.

Prices shown are subject to change without notice.
All orders must be prepaid.
In unusual cases, processing may exceed 30 days.

Circle 12 on inquiry card.

Listing 2, continued:

```
0950:  49 FF                 EOR   #$FF
0952:  F0 C4                 BEQ   BR1        BRANCH IF SO
0954:  60                    RTS
0955:  0A          BNM1      ASL   A          DOUBLE RESULT-REG INDEX
0956:  AA                    TAX
0957:  B5 00                 LDA   R0L,X
0959:  35 01                 AND   R0H,X      CHK BOTH BYTES FOR NO $FF
095B:  49 FF                 EOR   #$FF
095D:  D0 B9                 BNE   BR1        BRANCH IF NOT MINUS 1
095F:  60          NUL       RTS
0960:  A2 18       RS        LDX   #$18       12*2 FOR R12 AS STK PNTR
0962:  20 DD 08              JSR   DCR        DECR STACK POINTER
0965:  A1 00                 LDA   (R0L,X)    POP HIGH RETURN ADR TO PC
0967:  85 1F                 STA   R15H
0969:  20 DD 08              JSR   DCR        SAME FOR LOW-ORDER BYTE
096C:  A1 00                 LDA   (R0L,X)
096E:  85 1E                 STA   R15L
0970:  60                    RTS
0971:  4C 3E 08    RTN       JMP   RTNZ
                   *
                   * REG SAVE/RESTORE ROUTINES
                   * FOR NON-APPLE-II SYSTEMS
                   *
                   ASAV      EPZ   $45
                   XSAV      EPZ   $46
                   YSAV      EPZ   $47
                   PSAV      EPZ   $48
0974:  85 45       SAVE      STA   ASAV
0976:  86 46                 STX   XSAV       SAVE 6502 REG CONTENTS.
0978:  84 47                 STY   YSAV
097A:  08                    PHP
097B:  68                    PLA
097C:  85 48                 STA   PSAV
097E:  60                    RTS
097F:  A5 48       RESTORE   LDA   PSAV
0981:  48                    PHA
0982:  A5 45                 LDA   ASAV
0984:  A6 46                 LDX   XSAV       RESTORE 6502 REG CONTENTS.
0986:  A4 47                 LDY   YSAV
0988:  28                    PLP
0989:  60                    RTS
```

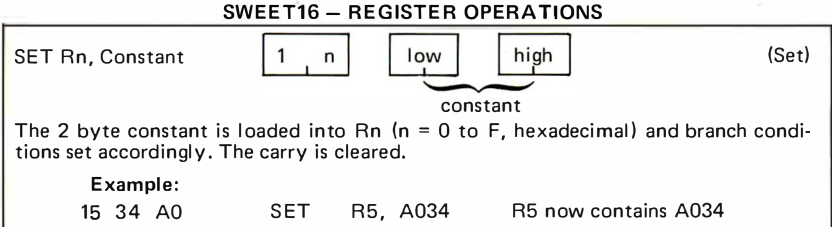
Table 1:

SWEET16 OP CODE SUMMARY

Register Ops				Nonregister Ops		
				00	RTN	(Return to 6502 mode)
1n	SET	Rn	Constant (Set)	01	BR ea	(Branch always)
2n	LD	Rn	(Load)	02	BNC ea	(Branch if No Carry)
3n	ST	Rn	(Store)	03	BC ea	(Branch if Carry)
4n	LD	@Rn	(Load indirect)	04	BP ea	(Branch if Plus)
5n	ST	@Rn	(Store indirect)	05	BM ea	(Branch if Minus)
6n	LDD	@Rn	(Load double indirect)	06	BZ ea	(Branch if Zero)
7n	STD	@Rn	(Store double indirect)	07	BNZ ea	(Branch if NonZero)
8n	POP	@Rn	(Pop indirect)	08	BM1 ea	(Branch if Minus 1)
9n	STP	@Rn	(Store pop indirect)	09	BNM1 ea	(Branch if Not Minus 1)
An	ADD	Rn	(Add)	0A	BK ea	(Break)
Bn	SUB	Rn	(Sub)	0B	RS	(Return from Subroutine)
Cn	POPD	@Rn	(Pop double indirect)	0C	BS ea	(Branch to Subroutine)
Dn	CPR	Rn	(Compare)	0D		(Unassigned)
En	INR	Rn	(Increment)	0E		(Unassigned)
Fn	DCR	Rn	(Decrement)	0F		(Unassigned)

SWEET16 Operation Code Summary: Table 1 summarizes the list of SWEET16 operation codes, which are explained in further detail one by one in the descriptions which follow the table. The program of listing 2 implements the execution of these interpretive codes after a call to the entry point **SW16**. Return to the calling program and normal noninterpretive operation is accomplished with the RTN mnemonic of SWEET16.

SWEET16 – REGISTER OPERATIONS



The 2 byte constant is loaded into Rn (n = 0 to F, hexadecimal) and branch conditions set accordingly. The carry is cleared.

Example:

```
15 34 A0      SET   R5, A034      R5 now contains A034
```

LD Rn | 2 n | (Load)

The ACC (R0) is loaded from Rn and branch conditions set according to the data transferred. The carry is cleared and the contents of Rn are not disturbed.

Example:

```
15 34 A0      SET   R5, A034
25            LD    R5            ACC now contains A034
```

in the specified register and can be sensed by subsequent branch instructions since the register specification is saved in the high order byte of R14. This specification is changed to indicate R0 (ACC) for ADD and SUB instructions and R13 for the CPR (compare) instruction.

Normally the high order R14 byte holds the "prior result register" index *times 2* to account for the 2 byte SWEET16 registers, and thus the least significant bit is zero. If ADD, SUB or CPR instructions generate carries, then this index is incremented, setting the least significant bit, which becomes a carry flag.

The SET instruction increments the program counter twice, picking up data bytes for the specified register. In accordance with 6502 convention, the low order data byte precedes the high order byte.

Most SWEET16 nonregister operations are relative branches. The corresponding subroutines determine whether or not the "prior result" meets the specified branch condition and if so update the SWEET16 program counter by adding the displacement value (−128 to +127 bytes).

The RTN operation restores the 6502 register contents, pops the subroutine return stack and jumps indirect through the SWEET16 program counter register. This transfers control to the 6502 at the instruction immediately following the RTN instruction.

The BK operation actually executes a 6502 break instruction (BRK), transferring control to the interrupt handler.

Any number of subroutine levels may be implemented within SWEET16 code via the BS (Branch to Subroutine) and RS (Return from Subroutine) instructions. The user must initialize and otherwise not disturb R12 if the SWEET16 subroutine capability is used since it is utilized as the automatic subroutine return stack pointer.

Memory Allocation and User Modifications

The only storage that must be allocated for SWEET16 variables are 32 consecutive locations in page zero for the SWEET16 registers, four locations to save the 6502 register contents, and a few levels of the 6502 subroutine return address stack. If you don't need to preserve the 6502 register contents, delete the SAVE and RESTORE subroutines and the corresponding subroutine calls. This will free the four page zero locations ASAV, XSAV, YSAV and PSAV.

You may wish to add some of your own

Text continued on page 159

Circle 81 on inquiry card.

ST Rn — 3 n — (Store)

The ACC (R0) is stored into Rn and branch conditions set according to the data transferred. The carry is cleared and the ACC contents are not disturbed.

Example:

```
25          LD    R5          Copy the contents
36          ST    R6          of R5 to R6.
```

LD @Rn — 4 n — (Load indirect)

The low order ACC byte is loaded from the memory location whose address resides in Rn, and the high order ACC byte is cleared. Branch conditions reflect the final ACC contents which will always be positive and never minus 1. The carry is cleared. After the transfer, Rn is incremented by 1.

Example:

```
15 34 A0    SET   R5, A034
45          LD   @R5          ACC is loaded from
                              memory location A034
                              and R5 is incremented
                              to A035.
```

ST @Rn — 5 n — (Store indirect)

The low order ACC byte is stored into the memory location whose address resides in Rn. Branch conditions reflect the 2 byte ACC contents. The carry is cleared. After the transfer, Rn is incremented by 1.

Example:

```
15 34 A0    SET   R5, A034    Load pointers R5 and R6
16 22 90    SET   R6, 9022    with A034 and 9022.
45          LD   @R5          Move a byte from location
56          ST   @R6          A034 to location 9022. Both
                              pointers are incremented.
```

LDD @Rn — 6 n — (Load double byte indirect)

The low order ACC byte is loaded from the memory location whose address resides in Rn, and Rn is then incremented by 1. The high order ACC byte is loaded from the memory location whose address resides in the (incremented) Rn and Rn is again incremented by 1. Branch conditions reflect the final ACC contents. The carry is cleared.

Example:

```
15 34 A0    SET   R5, A034
65          LDD  @R5          The low order ACC byte is
                              loaded from location A034,
                              the high order byte from
                              location A035. R5 is incre-
                              mented to A036.
```

STD @Rn — 7 n — (Store double byte indirect)

The low order ACC byte is stored into the memory location whose address resides in Rn, and Rn is then incremented by 1. The high order ACC byte is stored into the memory location whose address resides in (the incremented) Rn and Rn is again incremented by 1. Branch conditions reflect the ACC contents which are not disturbed. The carry is cleared.

Example:

```
15 34 A0    SET   R5, A034    Load pointers R5 and R6
16 22 90    SET   R6, 9022    with A034 and 9022. Move
65          LDD  @R5          double byte from locations
76          STD  @R6          A034 and A035 to locations
                              9022 and 9023. Both pointers
                              are incremented by 2.
```

POP @Rn | 8 n | (Pop indirect)

The low order ACC byte is loaded from the memory location whose address resides in Rn *after* Rn is decremented by 1 and the high order ACC byte is cleared. Branch conditions reflect the final 2 byte ACC contents which will always be positive and never minus 1. The carry is cleared. Because Rn is decremented prior to loading the ACC, single byte stacks may be implemented with the ST @Rn and POP @Rn operations (Rn is the stack pointer).

Example:

```
15 34 A0     SET   R5, A034    Init stack pointer.
10 04 00     SET   R0, 4          Load 4 into ACC.
35           ST   @R5          Push 4 onto stack.
10 05 00     SET   R0, 5          Load 5 into ACC.
35           ST   @R5          Push 5 onto stack.
10 06 00     SET   R0, 6          Load 6 into ACC.
35           ST   @R5          Push 6 onto stack.
85           POP  @R5          Pop 6 off stack into ACC.
85           POP  @R5          Pop 5 off stack.
85           POP  @R5          Pop 4 off stack.
```

STP @Rn | 9 n | (Store pop indirect)

The low order ACC byte is stored into the memory location whose address resides in Rn *after* Rn is decremented by 1. Then the high order ACC byte is stored into the memory location whose address resides in Rn after Rn is again decremented by 1. Branch conditions will reflect the 2 byte ACC contents which are not modified. STP @Rn and PLA @Rn are used together to move data blocks beginning at the greatest address and working down. Additionally, single byte stacks may be implemented with the STP @Rn and LDA @Rn ops.

Example:

```
14 34 A0     SET   R4, A034    Init pointers.
15 22 90     SET   R5, 9022
84           POP  @R4          Move byte from A033
95           STP  @R5             to 9021.
84           POP  @R4          Move byte from A032
95           STP  @R5             to 9020.
```

ADD Rn | A n | (Add)

The contents of Rn are added to the contents of the ACC (R0) and the low order 16 bits of the sum restored in ACC. The 17th sum bit becomes the carry and other branch conditions reflect the final ACC contents.

Example:

```
10 34 76     SET   R0, 7634    Init R0 (ACC)
11 27 42     SET   R1, 4227       and R1.
A1           ADD   R1          Add R1 (sum = B85B,
                                  carry clear)
A0           ADD   R0          Double ACC (R0) to 70B6
                                  with carry set.
```

SUB Rn | B n | (Subtract)

The contents of Rn are subtracted from the ACC contents by performing a two's complement addition:

ACC ← ACC + $\overline{Rn}$ + 1

The low order 16 bits of the subtraction are restored in the ACC. The 17th sum bit becomes the carry and other branch conditions reflect the final ACC contents. If the 16 bit unsigned ACC contents are greater than or equal to the 16 bit unsigned Rn contents then the carry is set, otherwise it is cleared. Rn is not disturbed.

Example:

```
10 34 76     SET   R0, 7634    Init R0 (ACC)
11 27 42     SET   R1, 4227       and R1.
A1           SUB   R1          Subtract R1 (diff = 340D
                                  with carry set)
A0           SUB   R0          Clears ACC (R0)
```



POPD @Rn [C n] (POP Double byte indirect)

Rn is decremented by 1 and the high order ACC byte is loaded from the memory location whose address now resides in Rn. Then Rn is again decremented by 1 and the low order ACC byte is loaded from the corresponding memory location. Branch conditions reflect the final ACC contents. The carry is cleared. Because Rn is decremented *prior* to loading each of the ACC halves, double byte stacks may be implemented with the STD @ Rn and POPD @ Rn operations. (Rn is the stack pointer).

Example:

```
15 34 A0        SET    R5, A034     Init stack pointer.
10 12 AA        SET    R0, AA12        Load AA12 into ACC.
75              STD   @R5           Push AA12 onto stack.
10 34 BB        SET    R0, BB34        Load BB34 into ACC.
75              STD   @R5           Push BB34 onto stack.
10 56 CC        SET    R0, CC56        Load CC56 into ACC.
75              STD   @R5
C5              POPD  @R5           Pop CC56 off stack.
C5              POPD  @R5           Pop BB34 off stack.
C5              POPD  @R5           Pop AA12 off stack.
```

CPR Rn [D n] (Compare)

The ACC (R0) contents are compared to Rn by performing the 16 bit binary subtraction ACC-Rn and storing the low order 16 difference bits in R13 for subsequent branch tests. If the 16 bit unsigned ACC contents are greater than or equal to the 16 bit unsigned Rn contents then the carry is set, otherwise it is cleared. No other registers, including ACC and Rn, are disturbed.

Example:

```
15 34 A0        SET    R5, A034     Pointer to memory.
16 BF A0        SET    R6, A0BF     Limit address.
10 00 00  LOOP  SET    R0, 0        Zero data.
75              STD   @R5           Clear 2 locs, incr R5 by 2.
25              LD     R5           Compare pointer R5
D6              CPR    R6              to limit R6.
02 F8           BNC    LOOP         Loop if carry clear.
```

INR Rn [E n] (Increment)

The contents of Rn are incremented by 1. The carry is cleared and other branch conditions reflect the incremented value.

Example:

```
15 34 A0        SET    R5, A034     Init R5 (pointer)
10 00 00        SET    R0, 0        Zero to R0.
55              ST    @R5           Clears loc A034 and incrs
                                       R5 to A035.
E5              INR    R5           Incr R5 to A036
55              ST    @R5           Clears loc A036 (not A035)
```

DCR Rn [F n] (Decrement)

The contents of Rn are decremented by 1. The carry is cleared and other branch conditions reflect the decremented value.

Example: (Clear nine bytes beginning at loc A034)

```
15 34 A0        SET    R5, A034     Init pointer.
14 09 00        SET    R4, 9        Init count.
10 00 00        SET    R0, 0        Zero ACC.
55        LOOP  ST    @R5           Clear a mem byte.
F4              DCR    R4           Decr count.
07 FC           BNZ    LOOP         Loop until zero.
```

SWEET16 Nonregister Instructions

RTN [0 0] (Return to 6502 mode)

Control is returned to the 6502 and program execution continues at the location immediately following the RTN instruction. The 6502 registers and status conditions are restored to their original contents (prior entering SWEET16 mode).

BR ea — `0 1` `d d` — (Branch Always)

An effective address (ea) is calculated by adding the signed displacement byte (dd) to the program counter. The program counter contains the address of the instruction immediately *following* the BR, or the address of the BR operation plus 2. The displacement is a signed two's complement value from −128 to +127. Branch conditions are not changed. Note that effective address calculation is identical to that for 6502 relative branches.

Some examples:

```
dd = $80    ea = PC + 2 - 128
dd = $81    ea = PC + 2 - 127
dd = $FF    ea = PC + 2 - 1
dd = $00    ea = PC + 2 + 0
dd = $01    ea = PC + 2 + 1
dd = $7E    ea = PC + 2 + 126
dd = $7F    ea = PC + 2 + 127
```

Example:

```
$300: 01 50        BR  $352
```

BNC ea — `0 2` `d d` — (Branch if No Carry)

A branch to the effective address is taken only if the carry is clear, otherwise execution resumes as normal with the next instruction. Branch conditions are not changed.

BC ea — `0 3` `d d` — (Branch if Carry set)

A branch is effected only if the carry is set. Branch conditions are not changed.

BP ea — `0 4` `d d` — (Branch if Plus)

A branch is effected only if the prior "result" (or most recently transferred data) was positive. Branch conditions are not changed.

Example: (Clear mem from loc A034 to A03F)

```
15 34 A0        SET  R5, A034    Init pointer.
14 3F A0        SET  R4, A03F    Init limit.
10 00 00  LOOP  SET  R0, 0
55              ST   @R5         Clear mem byte, incr R5.
24              LD   R4          Compare limit to
D5              CPR  R5            pointer.
04 F8           BP   LOOP        Loop until done.
```

BM ea — `0 5` `d d` — (Branch if Minus)

A branch is effected only if the prior "result" was minus (negative, MSB = 1). Branch conditions are not changed.

BZ ea — `0 6` `d d` — (Branch if Zero)

A branch is effected only if the prior "result" was zero. Branch conditions are not changed.

BNZ ea — `0 7` `d d` — (Branch if NonZero)

A branch is effected only if the prior "result" was nonzero. Branch conditions are not changed.

BM1 ea — `0 8` `d d` — (Branch if Minus 1)

A branch is effected only if the prior "result" was minus 1 ($FFFF hexadecimal). Branch conditions are not changed.

BNM1 ea — `0 9` `d d` — (Branch if Not Minus 1)

A branch is effected only if the prior "result" was not minus 1 ($FFFF hexadecimal). Branch conditions are not changed.

Text continued from page 154

instructions to this implementation of SWEET16. If you use the unassigned op codes $0E and $0F, remember that SWEET16 treats these as 2 byte instructions. You may wish to handle the break instruction as a SWEET16 call, saving two bytes of code each time you transfer into SWEET16 mode. Or you may wish to use the SWEET16 BK (Break) operation as a "CHAROUT" call in the interrupt handler. You can perform absolute jumps within SWEET16 by loading the ACC (R0) with the address you wish to jump to (minus 1) and executing a ST R15 instruction.

And as a final thought, the ultimate modification for those who do not use the 6502 processor would be to implement a version of SWEET16 for some other microprocessor design. The idea of a low level interpretive processor can be fruitfully implemented for a number of purposes, and achieves a limited sort of machine independence for the interpretive execution strings. I found this technique most useful for the implementation of much of the software of the Apple II computer; I leave it to readers to explore further possibilities for SWEET16.■

BRK | 0 | A | (Break)

A 6502 BRK (break) instruction is executed. SWEET16 may be reentered nondestructively at SW16D after correcting the stack pointer to its value prior to executing the BRK.

RS | 0 | B | (Return from SWEET16 Subroutine)

RS terminates execution of a SWEET16 subroutine and returns to the SWEET16 calling program which resumes execution (in SWEET16 mode). R12, which is the SWEET16 subroutine return stack pointer, is decremented twice. Branch conditions are not changed.

BS ea | 0 | C | | d | d | Branch to SWEET16 Subroutine

A branch to the effective address (PC + 2 + d) is taken and execution is resumed in SWEET16 mode. The current PC is pushed onto a "SWEET16 subroutine return address" stack whose pointer is R12, and R12 is incremented by 2. The carry is cleared and branch conditions set to indicate the current ACC contents.

Example: (Calling a "memory move" subroutine to move A034–A03B to 3000–3007)

```
300: 15 34 A0         SET   R5, A034    Init pointer 1.
303: 14 3B A0         SET   R4, A03B    Init limit 1.
306: 16 00 30         SET   R6, 3000    Init pointer 2.
309: 0C 15            BS    MOVE        Call move subroutine.
      .
      :
320: 45         MOVE  LD    @R5         Move one
321: 56               ST    @R6           byte.
322: 24               LD    R4
323: D5               CPR   R5          Test if done.
324: 04 FA            BP    MOVE
326: 0B               RS                Return.
```

Switching ROMs in the Fairchild F8 Evaluation Kit

John C Polonchak
GTE Sylvania
Electronic Components Group
114 S Oregon St
El Paso TX 79901

Many people who use the Fairchild F8 Evaluation Kit supplied with the Fairbug Monitor would like to try the Mostek DDT Monitor. Unfortunately, the read only memory chip containing the DDT Monitor cannot be directly substituted into the available socket as wired for the Fairbug read only memory chip. The scheme shown below permits the use of either chip without modifications to the printed circuit board. The DDT monitor chip is plugged into a wire wrap socket. This socket is wired to a printed circuit socket that plugs into the Fairbug socket. Most of the wire wrap pins can be plugged directly into the socket in piggyback style. The pins marked NC are not connected and can be cut off of the wire wrap socket.

This same type of rewiring scheme must also be performed on Q5 which is a 7406, since the Mostek and Fairchild boards use mutually inverted signals. Although both chips use the same instructions they seem to be complements of each other to the outside world. Tables 1 and 2 show the interconnections for the two ICs. Note that this scheme of simulating one memory or processor integrated circuit with another similar, but pinout-incompatible, chip can be used quite generally.■

Table 1: These are the pin interconnections for the DDT monitor memory to the Fairbug socket. Of the 40 pins that need connecting, 23 of them can be plugged directly into the socket; 14 of them have their pins cut off and aren't used, leaving three pins that need to be rewired.

DDT READ ONLY MEMORY 3851

Top wire wrap socket containing DDT	Bottom printed circuit socket.	Top wire wrap socket containing DDT	Bottom printed circuit socket.
1	2	21	21
2	NC	22	22
3	3	23	NC
4	4	24	NC
5	2	25	NC
6	6	26	NC
7	7	27	27
8	8	28	28
9	9	29	NC
10	10	30	NC
11	11	31	NC
12	12	32	NC
13	13	33	33
14	14	34	34
15	15	35	NC
16	16	36	NC
17	17	37	NC
18	18	38	19
19	NC	39	39
20	NC	40	40

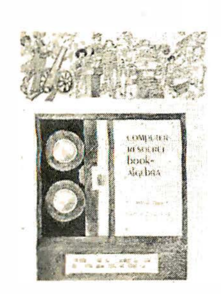
Q5 7406

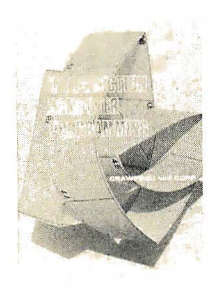
Top wire wrap socket containing 7406.	Bottom printed circuit socket.
1	1
2	4
3	NC
4	NC
5	5
6	8
7	7
8	NC
9	NC
10	10
11	11
12	12
13	13
14	14

Table 2: This is the rewiring diagram for the 7406 hexadecimal inverters. This chip must be rewired since the signals coming from the Fairbug monitor are complements of the signals from the Mostek DDT monitor.

A Bit of the BASIC

———**Computer Resource Book—Algebra** by Thomas A Dwyer and Margot Critchfield is an exciting new way to learn about algebra and the interesting things you can do with it using a computer. The book uses the BASIC language, and flowcharts are used throughout to show the structure of programs. There are 60 applications programs including straight line graphs, polynomial equations, a space probe navigator, temperature profiles, computer generated animation, the ultramatic root finder, random number generation and many more. Although it is particularly suitable for students, just about everyone will find some intriguing and easy to use applications in this entertaining book. $4.80.

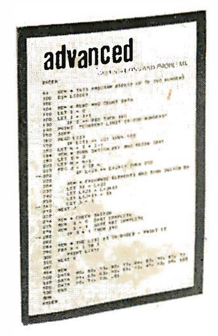




———**Introduction to Computer Programming** by Rudd A Crawford Jr and David H Copp. Here is an excellent way to learn about the general aspects of computer programming. **Introduction to Computer Programming** makes use of a hypothetical computer model and set of assembly language instructions designed to help the beginner see what goes on in computer programs. The emphasis throughout is on general principles; such concepts as loops, decisions, flowcharts and IO routines are covered in detail. The book also provides many example problems and prompts the reader by posing several quiz questions. Anyone who masters its contents will have a solid foundation for the study of practical assembly and high level languages. It is especially recommended for students, but just about everyone new to the subject should profit from it. $4.35.

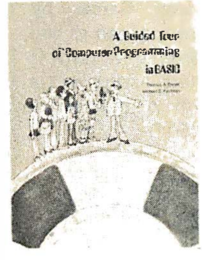
———**Basic BASIC** by James S Coan. If you're not already familiar with BASIC, James Coan's **Basic BASIC** is one of the best ways to learn about this popular computer language. BASIC (which stands for Beginner's All-purpose Symbolic Instruction Code) is easy to learn and easy to apply to many problems. **Basic BASIC** gives you step-by-step instructions for using a terminal, writing programs, using loops and lists, solving mathematical problems, understanding matrices and more. The book contains a wealth of illustrations and example programs, and is suitable for beginners at many different levels. It makes a fine reference for the experienced programmer, too. $7.95.

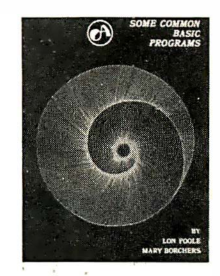




———**Advanced BASIC** by James S Coan. **Advanced BASIC** is the companion volume to James Coan's **Basic BASIC.** In this book you'll learn about some of the more advanced techniques for programming in BASIC, including string manipulation, the use of files, plotting on a terminal, simulation and games, advanced mathematical applications and more. Many useful algorithms are covered, including some clever sorting techniques designed to reduce program execution time. As with **Basic BASIC,** there are many illustrative example programs included. BASIC doesn't have to be basic with **Advanced BASIC**! $6.95.

———**A Guided Tour of Computer Programming in BASIC** by Thomas A Dwyer and Michael S Kaufman. Colorful graphics abound in this lively introduction to the BASIC language. The authors have tried to present a rigorous, yet entertaining approach to the subject. Written for the novice, **A Guided Tour** begins with a section on how to recognize a computer, followed by some tips on working at a terminal. By the end of the book readers are writing their own programs and solving elementary problems in finance and business. The emphasis throughout is on learning by doing. Anyone interested in computer programming should benefit from **A Guided Tour of Computer Programming in BASIC.** $4.80.





———**Some Common BASIC Programs** by Lon Poole and Mary Borchers, published by Adam Osborne and Associates. At last, a single source for all those hard to find mathematics programs! **Some Common BASIC Programs** combines a diversity of practical algorithms in one book: matrix multiplication, regression analysis, principal on a loan, integration by Simpson's rule, roots of equations, operations on two vectors, chi-square test, check writer, geometric mean and variation, coordinate conversion and a function plotting algorithm. These are just some of the many programs included. For only $7.50 you can buy the kind of programs previously available only as part of software math package systems for large scale computers. All the programs are written in a restricted BASIC suitable for most microcomputer BASIC packages, and have been tested and debugged by the authors. $7.50.

Send to:
Bits, Inc.
70 Main Street
Peterborough NH 03458

Check Payment method:
____My check is enclosed
____Bill my MC No.__________ Exp. date ______
____Bill my BAC No.__________ Exp. date ______

Name ____________________

Address ____________________

City ______ State ____ Zip Code ______

Signature ____________________

Total for all books checked $______

Postage, 50 cents per book for ____ books $______

Grand Total $______

Prices shown are subject to change without notice.
All orders must be prepaid.
In unusual cases, processing may exceed 30 days.

You may photocopy this page if you wish to leave your BYTE intact.

BYTE's Bits

An IEEE Microcomputer Course

The IEEE's PHP Group in the San Francisco area is sponsoring a series of talks entitled "Microcomputers: Practical Applications." This Thursday evening series, beginning November 3, will be tutorial in nature, focusing on such "how to" topics as: which microcomputer to use (from the user's viewpoint); architecture and memory; getting started, at work and at home. The series will run for five weeks, with registration set at $35 ($25 for IEEE members; $10 students), and will be in the Palo Alto and San Jose area. Contact David Guidici at Siltec, 3717 Haven Av, Menlo Park CA 94025, (415) 365-8600.■

Attention Canadian Readers... A First (Ever) Canadian Personal Computing Show

Personal Computing Showplace is the name of a show to be held in Toronto at the International Center of Commerce on November 8, 9 and 10 of this year. The show will feature the first (ever) exhibition of personal computing products in Canada.

The exhibit hall is located at 6900 Airport Rd in Toronto. Admission will be $5 at the door. Hours are 12 noon to 9 PM each day. This exhibition runs concurrently with the eighth annual Canadian Computer Show. For more information contact Betty Gray, at (416) 595-1811.■

An Opinion: SOFTWARE AND PATENTABILITY, 1977

While perusing some of the electronic data processing (EDP) literature of the past few years, it struck me that the subject of software patentability had been completely exhausted, and that another article on the subject would cause readers to shy away in revulsion. However, the ebullient EDP industry, in its constant state of change, has proferred a new development which makes all of the irrelevant jabberings of the past (some of them my own) now of considerable importance.

Why have I dismissed the plethora of past writings as irrelevant jabbering? Because the simple and direct answer to the question of whether software is patentable is "no!" Volumes have been filled discussing the point, yet it should have been apparent from the beginning that one insurmountable obstacle stood in the way: namely, that a patentable invention must have a physical existence and not be merely a methodology or mental process.

Hence we see the illogical result of hard wired programming being patentable but pure software not. But all of this really didn't make a great deal of difference because the industry was able to protect its investment in software development through other legal avenues such as trade secrecy and was therefore spared the labyrinthine procedures of the patent law.

All of this is soon to change. As microprogramming state of the art brings us closer to the point where a programmer can sit down with a few integrated circuit chips and a soldering iron rather than a pencil and coding sheet, his/her product (now a black box rather than a source listing) very clearly becomes a candidate for patentability because it is not just a mental process or algorithm any longer, but rather is a physical object.

This means that the arcane and convoluted laws of patents, together with those who administer them, suddenly become relevant in much the same way that a bull in a china shop is relevant. Let us examine a few of the more salient features of patent law and see how they might apply to microprogamming situations.

Novelty

An invention, both in Canada and the United States, must be a "*new* and useful art, process, machine, manufacture or composition of matter" to qualify for patent protection (emphasis mine). A new way of achieving a known result, or simply creating a new result, would permit an invention to qualify if the patent officer could be convinced of its novelty.

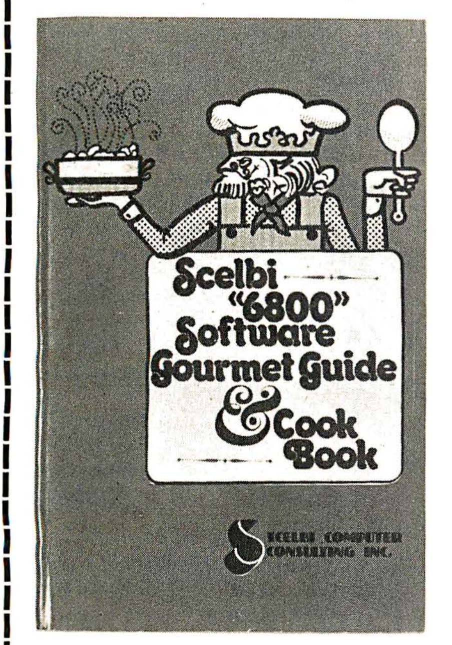
The reader can appreciate the Alice in Wonderland world which would soon be created by disputed claims revolving around the definition of "new."

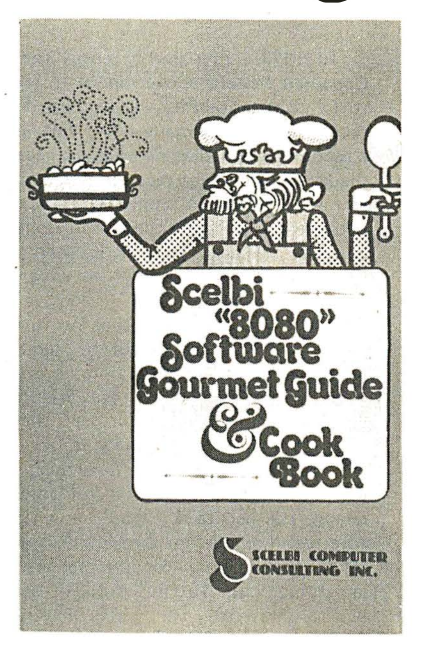
Expense

Because all of this jousting is done by high priced lawyers, the cost of obtaining a patent and prosecuting infringement quickly becomes prohibitive for all but the largest corporations; so does defending against an accusation of infringement.

This means that a belligerent patent holder is armed with a big legal stick, the

Gourmet Reading from BITS

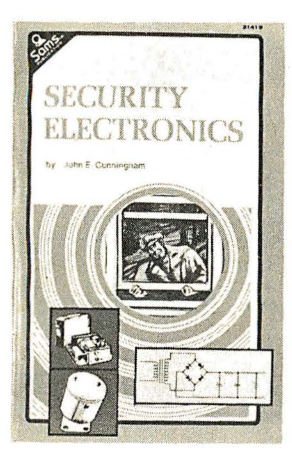


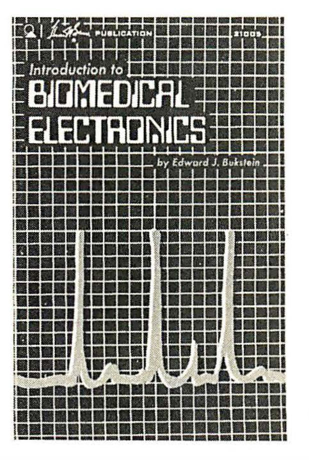


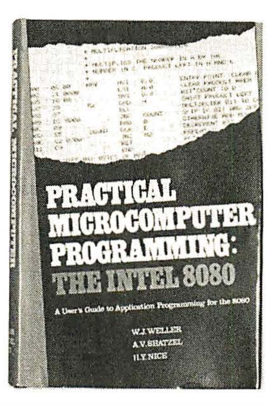
____ **Scelbi "6800" Software Gourmet Guide & Cookbook**

____ **Scelbi "8080" Software Gourmet Guide & Cookbook**, both by Robert Findley. Have you tried cooking up a program lately on your 6800 or 8080 processor? Have you needed a dash of ideas on how to add spice to a program? Then the **Scelbi "6800" Software Gourmet Guide & Cookbook** and the **Scelbi "8080" Software Gourmet Guide & Cookbook** may prove to be quite useful additions to your library.

Both books contain a wealth of information on the 6800 and 8080 instruction sets, plus general programming techniques applied to the 8080 and 6800, conversion routines, floating point routines, decimal arithmetic routines, and much more. Order yours today and get a taste of what your processor can do. Both books are priced at $9.95 each.







____ **Security Electronics** by John E Cunningham. To catch a thief, apply liberal doses of ingenuity and a modicum of cleverness. Find out what's been tried in conventional alarm systems before you go off computerizing your home security system, though. New 2nd edition $5.95

____ **Introduction to Biomedical Electronics** by Edward J Bukstein. What's been done in robot doctors? Nothing so far. But in terms of electronic aids to physicians and practices of health researchers, consult this background review of the field of biomedical electronics $5.95

____ **Practical Microcomputer Programming: The Intel 8080** by W J Weller, A V Shatzel, and H Y Nice. Here is a comprehensive source of programming information for the present or prospective user of the 8080 microcomputer, an architecture which appears in the MITS Altair, 8800, Processor Technology SOL, IMSAI 8080, Polymorphics POLY-88, and other popular microcomputer system products.

After several preliminary chapters, the authors get down to practical details with topics such as moving data, binary arithmetic operations, multiplication and division, use of the stack pointer, subroutines, arrays and tables, conversions, decimal arithmetic, various IO options, real time clocks and interrupt driven processes, and debugging techniques. Most examples are given in symbolic assembly form, with occasional listings of assembled code using a Computer Automation software development system.

This 306 page hardcover book is well worth its $21.95 price and should be in every 8080 or Z-80 user's library.

Send to:

BITS, Inc
70 Main St
Peterborough NH 03458

Check payment method:

___ My check is enclosed
___ Bill my MC No. ______________ Exp. date ______
___ Bill my BAC No. ______________ Exp. date ______

Total for all books checked $ ________
Postage, 50 cents per book for ________ books $ ________
Grand Total $ ________

Name ____________________
Address ____________________
City ____________ State ______ Zip ______
Signature ____________________

You may photocopy this page if you wish to leave your BYTE intact. In unusual cases, processing may exceed 30 days

Circle 12 on inquiry card.

use of which need only be threatened in order to put financially limited black box developers onto the ropes: either they fight a protracted and expensive legal battle or pull their product off the market.

In my own law practice, clients have sought advice and expressed their concern that software packages which they proposed to develop might tread on someone else's rights. I have advised them that there need be no such fear so long as misuse of trade secrets or breach of a confidentiality agreement is not involved. If black box patents begin to appear I, for one, will not continue to give the same advice in the future because inadvertent duplication through independent invention is no excuse in a patent matter.

By contrast, if the system architecture of a piece of software is a trade secret, anyone who independently develops a like product is free to market it. However, the same piece of software, if embodied in a black box and protected by a patent, is the exclusive preserve of the patent holder, and anyone independently developing a like black box would risk infringement unless a costly and time consuming search of the patent records was made beforehand.

A Hands-Off Policy

In 1971 I prepared a report for the Canadian Federal Government in which software patentability was examined and recommendation made. At that time I recommended that no changes need be made to the Patent Act so as to extend its application to computer software because the Act would prove inadequate and also because the EDP industry had found trade secrecy laws to meet its needs reasonably well.

I also recommended that no further study in the area would be required until the nature of programming underwent a fundamental change. Although I cannot claim to have foreseen the advent of microprogramming (much less the astonishing speed of its arrival) one can now say that such a fundamental change has occurred. I would now therefore recommend that a review be undertaken immediately for the purpose of amending the Patent Act so as to exclude black box microprogramming from its operation.

Such heresy will probably be rejected by bureaucrats who see a golden opportunity to expand their little kingdoms and also by the many patent law firms which would stand to do a land office business.

However, it is accepted that the Patent Act has had serious limitations even in those traditional areas of industry for which it was originally designed. How much less useful it would be in the world of EDP is made plain by that industry's general avoidance of patent law wherever possible. That being so, it is this writer's view that the industry should not be dragged kicking and screaming into a morass which it has until now successfully sidestepped.

Daniel A Mersich, Attorney
1262 Don Mills Rd, Suite 17
Don Mills, Ontario
CANADA

[Daniel Mersich is a Toronto lawyer whose practice is restricted to EDP matters. . .CM] ■

Continued from page 37

is part of a feature that allows statements numbers to be interpreted.). Readers are welcome to contact me for further information on this interpreter.

I found one additional error: In the right center section of figure 5 on page 61, the second line in the box above the ERROR 2 terminal should read "+S(J) /10 COUNT".

My thanks to Mr Dickey and to the many other eagle-eyed readers who wrote in to report these errors. ■

A Revision to "Using a Keyboard ROM"

→

The article on using a KR2376 keyboard encoder ROM in the May 1977 BYTE, page 76, would have been a great help rather than a hindrance had the author supplied the connections for the standard version of this device (as was claimed) in his figure 2. I offer a revised figure which agrees with the code assignment charts suppied in both the GI and SMC data sheets and, I belatedly discovered, Don Lancaster's *TVT Cookbook*. There is very little in common between the two charts. The author was obviously working with a nonstandard unit.

Dr Samuel I Green
13052 Ferntrails Ln
Creve Coeur MO 63141

This is confirmed by a communication from GI, who informs us that the unit by Mr Brehm was indeed a surplus part obtained from a manufacturer of custom encoded keyboards. ■

		X_0	X_1	X_2	X_3	X_4	X_5	X_6	X_7
Y_0	N	NUL	DLE	—	ϕ	;	l	o	9
	S	NUL	DLE	=	NUL	+	L	O	)
	C	NUL	DLE	NUL	NUL	NUL	FF	SI	NUL
Y_1	N	SCH	K	FS	:	/	k	i	8
	S	SCH	[	FS	*	?	K	I	(
	C	SCH	VT	FS	NUL	NUL	VT	HT	NUL
Y_2	N	STX	L	GS	P	.	j	u	7
	S	STX	\	GS	P	>	J	U	/
	C	STX	FF	GS	DLE	NUL	LF	NAK	NUL
Y_3	N	ETX	~	RS	—	'	h	y	6
	S	ETX	Λ	RS	DEL	<	H	Y	&
	C	ETX	SO	RS	US	NUL	BS	EM	NUL
Y_4	N	EOT	M	US	@	m	g	t	5
	S	EOT	]	US	\	M	G	T	%
	C	EOT	CR	US	NUL	CR	BEL	DC4	NUL
Y_5	N	ENQ	NAK	<	BS	n	f	r	4
	S	ENQ	NAK	<	BS	N	F	R	$
	C	ENQ	NAK	NUL	BS	SO	ACK	DC2	NUL
Y_6	N	ACK	SYN	>	[	b	d	e	3
	S	ACK	SYN	>	(	B	D	E	#
	C	ACK	SYN	NUL	ESC	STX	EOT	ENQ	WUL
Y_7	N	BEL	ETB	,	]	v	s	w	2
	S	BEL	ETB	,	)	V	S	W	"
	C	BEL	ETB	NUL	GS	SYN	DC3	ETB	NUL
Y_8	N	DC1	CAN	SP	CR	c	a	q	1
	S	DC1	CAN	SP	CR	C	A	Q	!
	C	DC1	CAN	SP	CR	ETX	SOH	DC1	NUL
Y_9	N	P	EM	.	LF	x	FF	HT	Λ
	S	@	EM	.	LF	X	FF	HT	~
	C	DLE	EM	NUL	LF	CAN	FF	HT	RS
Y_{10}	N	0	SUB	—	DEL	z	ESC	VT	\
	S	—	SUB	—	DEL	Z	ESC	VT	:
	C	SI	SUB	US	DEL	SUB	ESC	VT	FS

N = Normal
S = Shift
C = Control

extensys corporation **64K for $1495**

Available now at these leading dealers

ALABAMA

COMPUTER CENTER
433 Valley Ave. Plaza
Birmingham, AL 35209
(205) 942-8567

COMPUTERLAND OF HUNTSVILLE
3020 University Drive N.W.
Huntsville, AL 35805
(205) 539-1200

ARIZONA

BYTE SHOP OF ARIZONA
813 N. Scottsdale Road
Tempe, AZ 85282
(602) 894-1129

BYTE SHOP OF PHOENIX N.W.
12654 N. 28th Drive
Phoenix, AZ 85019
(602) 942-7300

BYTE SHOP OF PHOENIX
28 W. Camelback Rd.
Phoenix, AZ 85013
(602) 942-7300

BYTE SHOP OF TUCSON
2612 E. Broadway
Tucson, AZ 85716
(602) 327-4579

MICRO AGE
803 N. Scottsdale Blvd.
Tempe, AZ 85281
(602) 894-1193

CALIFORNIA

BYTE SHOP OF CITRUS HEIGHTS
6041 Greenback Lane
Citrus Heights, CA 95610
(916) 961-2983

BYTE SHOP OF DIABLO VALLEY
2989 N. Main Street
Walnut Creek, CA 94596
(415) 933-6252

BYTE SHOP OF FRESNO
3139 E. McKinley Avenue
Fresno, CA 93703
(209) 485-2417

BYTE SHOP OF HAYWARD
1122 B Street
Hayward, CA 94541
(415) 537-2983

BYTE SHOP OF LAWNDALE
16508 Hawthorne Blvd.
Lawndale, CA 90260
(213) 371-2421

BYTE SHOP OF SAN JOSE
2626 Union Avenue
San Jose, CA 95124
(408) 377-4685

BYTE SHOP OF SAN RAFAEL
509 Francisco Blvd.
San Rafael, CA 94901
(415) 457-9311

BYTE SHOP OF SANTA CLARA
3400 El Camino Real
Santa Clara, CA 95050
(408) 249-4221

CALIFORNIA MICROCOMPUTERS
329 Flume
Chico, CA 95927
(916) 891-1730

COMPUTER ELECTRONICS
905 Delavina
Santa Barbara, CA 93101
(805) 962-4198

COMPUTER POWER & LIGHT
12321 Ventura Blvd.
Studio City, CA 91604
(213) 760-0405

COMPUTER ROOM
124H Blossom Hill Road
San Jose, CA 95123
(408) 226-8383

COMPUTER STORE
820 Broadway
Santa Monica, CA 90401
(213) 451-0713

COMPUTER STORE OF S.F.
1093 Mission Street
San Francisco, CA 94103
(415) 431-0640

COMPUTERLAND CORPORATION
1922 Republic Avenue
San Leandro, CA 94577
(415) 895-9363

COMPUTERLAND OF EL CERRITO
11074 San Pablo Avenue
El Cerrito, CA 94530
(415) 233-5010

COMPUTERLAND OF HAYWARD
22634 Foothill Blvd.
Hayward, CA 94542
(415) 538-8080

COMPUTERLAND OF SADDLEBACK VALLEY
24001 Via Fabricante #904
Mission Viejo, CA 92675
(714) 770-0131

COMPUTERLAND OF S.F.
117 Fremont Street
San Francisco, CA 94105
(415) 546-1592

COMPUTERLAND OF SAN DIEGO
4233 Convoy Street
San Diego, CA 92111
(714) 560-9912

COMPUTERLAND OF THOUSAND OAKS
El Cid Plaza
171 E. Thousand Oaks Blvd.
Suite 105
Thousand Oaks, CA 91360
(805) 497-9595

COMPUTERLAND OF TUSTIN
104 W. First Street
Tustin, CA 92680
(714) 544-0542

COMPUTERLAND OF WEST L.A.
Computers Unlimited, Inc.
6840 La Cienega Blvd.
Inglewood, CA 90302
(213) 776-8080

DIGITAL DELI
80 W. El Camino Real
Mt. View, CA 94040
(415) 961-2670

PEOPLE'S COMPUTER SHOP
13452 Ventura Blvd.
Sherman Oaks, CA 91423
(213) 789-7514

POLYMORPHIC, INC.
460 Ward Drive
Santa Barbara, CA 93111
(805) 967-2351

PROKOTRONICS
439 Marsh Street
San Luis Obispo, CA 93401
(805) 544-5441

RECREATIONAL COMPUTER CENTER
1324 S. Mary Avenue
Sunnyvale, CA 94087
(408) 735-7480

SUNSHINE COMPUTER COMPANY
20710 S. Leapwood
Carson, CA 90746
(213) 327-2118

XIMEDIA
1290 24th Avenue
San Francisco, CA 94122
(415) 566-7472

COLORADO

BYTE SHOP OF BOULDER
Palmer Gardens Shopping Center
3101 Walnut Street
Boulder, CO 80301
(303) 444-6550

FLORIDA

BYTE SHOP OF FT. LAUDERDALE
1044 E. Oakland Park Blvd.
Ft. Lauderdale, FL 33334
(305) 561-2983

BYTE SHOP OF MIAMI
7825 Bird Road
Miami, FL 33155
(305) 264-2983

DELTA ELECTRONICS
2000 Highway 441 E.
Leesburg, FL 32748
(904) 728-2478

ECONOMY COMPUTING SYSTEMS
2200 Forsyth Road
Orlando, FL 32807
(305) 678-4225

MICROCOMPUTER SYSTEMS, INC.
144 S. Dale Mabry Highway
Tampa, FL 33609
(813) 879-4301 or 879-4225

COMPUTER AGE
999 S.W. 40th Avenue
Plantation, FL 33317
(305) 791-8080

GEORGIA

ATLANTA COMPUTER MART
5091 Buford Highway
Atlanta, GA 30340
(404) 455-0647

ILLINOIS

ASPEN COMPUTERS, INC.
7521 W. Irving Park Road
Chicago, IL 60634
(312) 589-1212
(312) 625-3555/6

COMPUTERLAND OF ARLINGTON HEIGHTS
50 East Rand Road
Arlington Heights, IL 60004
(312) 255-6488

COMPUTERLAND OF NILES
9511 N. Milwaukee Avenue
Niles, IL 60648
(312) 967-1714

ITTY-BITTY MACHINE CO.
1322 Chicago Avenue
Evanston, IL 60201
(312) 328-6800

THE NUMBERS RACKET
623-1/2 S. Wright Street
Champaign, IL 61820
(217) 352-5435

DATA DOMAIN OF SCHAUMBURG
1612 E. Algonquin Road
Schaumburg, IL 60195
(312) 397-8700

INDIANA

AUDIO SPECIALISTS
415 N. Michigan
South Bend, IN 46601
(219) 234-5001

DATA DOMAIN OF BLOOMINGTON (HOME OFFICE)
406 S. College Ave.
Bloomington, IN 47401
(812) 334-3607

DATA DOMAIN OF INDIANAPOLIS
7027 Northwestern Avenue
Indianapolis, IN 46268
(317) 251-3139

DATA DOMAIN OF FORT WAYNE
2805 East State Blvd.
Fort Wayne, IN 46805
(219) 484-7611

HOME COMPUTER CENTER
2115 E. 62nd Street
Indianapolis, IN 46220
(317) 251-6800

DATA DOMAIN OF WEST LAFAYETTE
219 W. Columbia
West Lafayette, IN 47906
(317) 943-3951

KENTUCKY

COMPUTERLAND OF LOUISVILLE
813 B Lyndon Lane
Louisville, KY 40222
(502) 425-8308

DATA DOMAIN OF LOUISVILLE
3028 Hunsinger Lane
Louisville, KY 40220
(502) 456-5242

DATA DOMAIN OF LEXINGTON
506½ Euclid Avenue
Lexington, KY 40501
(606) 233-3346

MARYLAND

COMPUTER WORKSHOP
1776 E. Jefferson Street
Rockville, MD 20852
(301) 468-0455

COMPUTERLAND OF GAITHERSBURG
16065 Frederick Road (Rt. 355)
Rockville, MD 20855
(301) 948-7676

MASSACHUSETTS

THE CPU SHOP
39 Pleasant Street
Charlestown, MA 02109
(617) 241-9556

MINNESOTA

COMPUTER DEPOT, INC.
3515 W. 70 Street
Minneapolis, MN 55435
(612) 927-5601

MINNESOTA COMPUTERS INC.
7710 Computer Avenue
Suite 132
Edina, MN 55435
(612) 835-3850

MISSOURI

K & K COMPANY
15 East 31 Street
Kansas City, MO 64108
(716) 561-1776

NEBRASKA

OMAHA COMPUTER STORE
4540 South 84th Street
Omaha, NB 68127
(402) 592-3590

NEW JERSEY

COMPUTER MART OF NEW JERSEY
501 Highway 27
Iselin, NJ 08830
(201) 283-0600

COMPUTERLAND OF MORRISTOWN
2 DeHart Street
Morristown, NJ 07960
(201) 539-4077

HOBOKEN COMPUTER WORKS
20 Hudson Place
Hoboken, NJ 07030
(201) 420-1644

NEW YORK

COMPUTER CORNER
200 Hamilton Ave.
White Plains, NY 10601
(914) 949-3282

COMPUTERLAND OF BUFFALO
1612 Niagara Falls Blvd.
Buffalo, NY 14150
(716) 836-6511

COMPUTERLAND OF ITHACA
225 Elmira Road
Ithaca, NY 14850
(607) 277-4888

NORTH CAROLINA

COMPUTER ROOM
1729 Garden Terrace
Charlotte, NC 28203
(704) 377-9821

OHIO

BYTE SHOP OF OHIO
19524 Center Ridge Road
Rocky River, OH 44116
(216) 333-3261

DATA DOMAIN OF DAYTON
1932 Brown Street
Dayton, OH 45409
(513) 223-2348

DATA DOMAIN OF CINCINNATI
7694 Camargo Road
Cincinnati, OH 45243
(513) 561-6733

OKLAHOMA

HIGH TECHNOLOGY
1020 W. Wilshire Blvd.
Oklahoma City, OK 74116
(405) 842-2021

OREGON

ALTAIR COMPUTER CENTER
8105 Nimbus Avenue N.W.
Beaverton, OR 97005
(503) 644-2314

BYTE SHOP OF PORTLAND
2033 S.W. 4th Avenue
Portland, OR 97201
(503) 223-3496

REAL OREGON COMPUTER CO.
205 W. 10th Street
Eugene, OR 97401
(503) 484-1040

PENNSYLVANIA

PERSONAL COMPUTER CORP.
Frazer Mall
Malvern, PA 19355
(215) 647-8463

RHODE ISLAND

COMPUTER POWER, INC.
M24 Airport Mall
1800 Post Road
Warwick, RI 02886
(401) 738-4477

SOUTH CAROLINA

COMPUTER COMPANY
73 State Street
Charleston, SC 29401

TEXAS

COMPUTER SHOPS, INC.
211 Keystone Park
13933 N. Central Expwy.
Dallas, TX 75243
(214) 234-3412

COMPUTERLAND OF AUSTIN
Shoal Creek Plaza
3300 Anderson Lane
Austin, TX 78757

COMPUTERLAND OF S.W. HOUSTON
6439 Westheimer
Houston, TX 77057
(713) 977-0909/0910

K.A. ELECTRONICS
1220 Majesty Drive
Dallas, TX 75247
(214) 634-7870

THE MOS
1853 Richmond Avenue
Houston, TX 77098
(713) 527-8008

YOUNG ELECTRONIC SERVICE
P.O. Box DD
College Station, TX 77840
(713) 693-3462

UTAH

MICRODATA SYSTEMS
796 E. Lazon Drive
Sandy, UT 84070
(801) 561-4665

WASHINGTON

KBC COMPUTER SHOP
P.O. Box 169
Redmond, WA 98052
(206) 885-1694

RETAIL COMPUTER STORE
410 N.E. 72nd
Seattle, WA 98115
(206) 524-4101

WEST VIRGINIA

ALLEGHANY COMPUTER SERVICES
P.O. Box 767
Elkins, WV 26241
(304) 636-6600

WISCONSIN

MILWAUKEE COMPUTER STORE
6916 W. North Ave.
Milwaukee, WI 53213
(414) 259-9140

extensys corporation
592 Weddell Drive
Sunnyvale, California 94086
(408) 734-1525

SEE FULL PAGE AD ON PAGE 93 FOR DETAILS.
Circle 58 on inquiry card.

Do You Need the Real Time?

Gregory A R Trollope
433 Cherry Ln
Lewiston NY 14092

There are a number of ways of implementing a real time clock for a microcomputer system. With the many different clock chips available on the market, it would seem natural to try to interface to one of these. If the requirements set for a real time clock design are: the computer should be able to read the clock when necessary, the clock should keep time while other programs are executing, and one should be able to set the clock by computer commands, then use of external hardware can be a most difficult challenge. Taking a software approach using minimal hardware can be a most attractive alternative. In the implementation described here, the computer itself counts 1/10 second pulses, derived from the 50/60 Hz line, in four memory bytes, one each for hours, minutes, seconds and deciseconds. Setting the time becomes a process of writing the correct time to memory; reading the time, one of reading memory.

To enable other programs to execute while the clock is being maintained, the computer is forced into the clock counting routine only on the arrival of each 1/10 second pulse, and stays in the routine for only as long as necessary to perform the rather simple calculations before returning to the program that was executing. In my version this is achieved by using the interrupt request line of the 6800 processor, although it is conceivable that the NMI (nonmaskable interrupt) of the 6800 could be used as well.

Pin 4 of the 6800 processor is designated the $\overline{IRQ}$ line. When this line has a transition from logic 1 (~ +5 V) to logic 0 (ground), the processor finishes its current instruction, stacks all the registers and the address of the next instruction, then branches to the memory location contained in the memory bytes that respond to hexadecimal addresses FFF8 and FFF9. In systems using Motorola's MIKBUG monitor, such as the SwTPC 6800 computer, the interrupt vector addresses are in MIKBUG and point to another address in MIKBUG so that control of the processor is passed to MIKBUG after the interrupt has happened. MIKBUG dutifully passes control on by branching to the address contained in its volatile user memory at hexadecimal locations A000 and A001. If this address happens to be that of clock counting routine, it will receive control. Since all the registers and machine states have been saved, we can use them to do the

```
      REAL TIME CLØCK FØR MIKBUG SYSTEMS

                  *
                  *
                  ***GENERAL INSTRUCTIØNS***
                  *
                  *   AFTER LØADING THIS REAL TIME CLØCK PACKAGE,
                  *   EXIT MIKBUG WITH THE G CØMMAND, THEREBY
                  *   ENTERING THE CLØCK INITIALISATIØN RØUTINE.
                  *   THE RØUTINE WILL RETURN TØ MIKBUG;  TØ GØ TØ
                  *   THE DEMØNSTRATIØN, TYPE G AGAIN.'  THIS WILL
                  *   PRINT THE TIME EVERY 1 ØR 2 SECØNDS.
                  *   TØ EXIT, HIT THE BREAK KEY (ØR ANY ØTHER KEY,
                  *   SEVERAL TIMES).  TYPING G WILL RETURN TØ THE
                  *   DEMØNSTRATIØN.
0700                     ØRG   A $700
                  *
                  ***CLØCK CØUNTERS
0700 00           HR     FCB   A 0   CØNTAINS (24-HØURS)
0701 00           MIN    FCB   A 0            (60-MINS)
0702 00           SEC    FCB   A 0            (60-SECS)
0703 00           SPLIT  FCB   A 0            (10-DECISECS)
                  *
                  *
                  ***CLØCK INITIALISATIØN RØUTINE***
                  *
                  *       BRANCH HERE MANUALLY TØ SET CLØCK
                  *
                  * ENTER THE TIME IN THE FØRMAT
                  *       HHMM
                  *E.G.  1803     FØR 3 MINS PAST 6PM
                  *ENTER THE LAST DIGIT ØN THE TIME SIGNAL
                  *
0704 0F           CLØCK  SEI            STØP CLØCK
0705 86 07               LDA A # 7      CØNFIGURE SWTPC PØRT 7
0707 B7 801F             STA A E $801F AS THE CLØCK INTERFACE
                  *
070A 8D 19               BSR   R IN2    GET HH
070C 86 18               LDA A # 24
070E 10                  SBA
070F B7 0700             STA A E HR    SET HØURS
0712 8D 11               BSR   R IN2    GET MM
0714 86 3C               LDA A # 60
0716 B7 0702             STA A E SEC    SET SECS TØ 0
0719 10                  SBA
071A B7 0701             STA A E MIN    SET MINS
071D 86 0A               LDA A # 10
071F B7 0703             STA A E SPLIT  SET SPLITS
0722 0E                  CLI            FREE CLØCK
0723 20 68               BRA   R D2
                  *
0725 BD E0AA      IN2    JSR   E $E0AA  'INHEX' FIRST DIGIT
0728 48                  ASL A          H*2
0729 16                  TAB
```

Listing 1a: The real time clock software for a 6800 system which uses an IRQ interrupt input from a PIA port. This routine is intended to be used with the Motorola MIKBUG software, and includes provision for setting time of day in hours, minutes and seconds.

Note: *This assembler uses a format in which explicit indication of address type is indicated where one mnemonic has several possibilities. Thus, for example, LDA A # 7 means use immediate (#) addressing of the operand 7, while LDA A E 7 means use extended addressing (E) of the operand at memory location 7. Other abbreviations seen in this listing are R for relative addressing (branches only), I for immediate 2 byte operand; and A designates an assembler directive.*

counting without worrying. When we have done all that needs to be done, the RTI instruction restores the registers and branches back to the program that was interrupted.

But how do we interface a signal to IRQ? The simplest way might be to connect it directly, but if one wishes to preserve the option of finding out what caused the interrupt, some additional logic is necessary. While there are probably all sorts of clever ways of doing this, a convenient way to implement it is by using a peripheral interface adapter (PIA). The SwTPC 6800 computer which I use has a parallel interface card which has a PIA. This has 16 data lines and four control lines. We will use one of the control lines, CB1, to latch the clock pulses and provide input conditioning of the interrupt signal. Part of the clock setting routine must be to configure the PIA properly, and part of the interrupt routine to acknowledge the interrupt, which is achieved by reading the PIA B data register, but more of the details later.

A self-supporting real time clock package is given in listing 1b. The package is assembled at hexadecimal location 0700 so as to

The more astute students of MIKBUG will notice that the control line CA1 of the MIKBUG serial control PIA is configured by MIKBUG to cause an interrupt on its transition; CA1 is left free in the SwTPC serial control interface; and interrupts can be passed to the IRQ line. It is possible to implement the clock, then, by routing the clock pulses to this MIKBUG oriented PIA, with one proviso: that the interface not be used for IO! It is an idiosyncrasy of the PIA that if you happen to be reading the data register when the interrupt occurs, the IRQ bit in the control register will not be set, even though the interrupt routine will be entered ~ 99.99...% of the time. Thus if MIKBUG is doing its IO at the same time as the clock pulse occurs, there is a small, but nonzero chance, that some interrupts will be lost, causing long term timing inaccuracies for a continuously running real time clock.

```
S00600004844521B
S1150700000000000F8607B7801F8D19861810B70700DF
S11307123D11B63CB7070210B707018 60AB70703 93
S113072 20E2068BDE0AA48164848 1B16BDE0AA1B65
S1050732163972
S1150734B6801F2A29B6801E7A07032621860AB7070397
S11507467A07022617863CB707027A0701260DB70701E7
S11307587A07002605861 8B70700 3BC618F0070075
S113076 8D2FC63CF007018D28C63CF007028D2169
S1140778CEE19DBDE07EB60702F680042A07B10702E1
S11307892 7F620D6CE0763FFA0487FA0437EE0E387
S11407994FC10A2B068B10C00A20F61B3630BDE0CA9D
S10507AA3139DF
S105A0000734 1F
S104A0430018
S105A04807040 7
```

Listing 1b: Object code listing in MIKBUG format for the real time clock program of figure 1a.

Listing 1a, continued:

```
072A 48                ASL A          H*4
072B 48                ASL A          H*8
072C 1B                ABA            H*8+H*2=H*10
072D 16                TAB             SAVE
072E BD E0AA           JSR   E $E0AA   SECØND DIGIT
0731 1B                ABA             FØRM SUM
0732 16                TAB             SAVE
0733 39                RTS
                *
                *
                ***CLØCK CØUNTING RØUTINE***
                *
                *    IRQ BRANCHES HERE
                *
0734            IRQ    RMB   A 0
                *
                *(INSERT PØLLS TØ ANY ØTHER ITERRUPTS)
                *
0734 B6 801F           LDA A E $801F  IRQ FRØM CLØCK INTERFACE?
0737 2A 29             BPL   R RTI   BRANCH IF NØT
0739 B6 801E           LDA A E $801E   CLEAR INTERRUPT
073C 7A 0703           DEC   E SPLIT DECREMENT CØUNTER
073F 26 21             BNE   R RTI   BRANCH IF NØT ZERØ
0741 86 0A             LDA A # 10     RESET
0743 B7 0703           STA A E SPLIT
0746 7A 0702           DEC   E SEC    DECREMENT NEXT CØUNTER
0749 26 17             BNE   R RTI    ...ETC
074B 86 3C             LDA A # 60
074D B7 0702           STA A E SEC
0750 7A 0701           DEC   E MIN
0753 26 0D             BNE   R RTI
0755 B7 0701           STA A E MIN
0758 7A 0700           DEC   E HR
075B 26 05             BNE   R RTI
075D 86 18             LDA A # 24
075F B7 0700           STA A E HR
0762 3B         RTI    RTI
                *
                *
                ***CLØCK DEMØNSTRATIØN***
                *
                *   BRANCH HERE MANUALLY
                *
0763 C6 18      DEMØ   LDA B # 24     CØMPLEMENT
0765 F0 0700           SUB B E HR     HØURS
0768 8D 2F             BSR   R ØUT     PRINT
076A C6 3C             LDA B # 60     MINS
076C F0 0701           SUB B E MIN
076F 8D 28             BSR   R ØUT     PRINT
0771 C6 3C             LDA B # 60     SECS
0773 F0 0702           SUB B E SEC
0776 8D 21             BSR   R ØUT     PRINT
                *
0778 CE E19D           LDX   I $E19D  'MCL' ... SEND CR/LF
077B BD E07E           JSR   E $E07E    'PDATA1'
                *
077E 86 0702           LDA A E SEC    SAVE TIME
0781 F6 8004    D1     LDA B E $8004    TEST FØR BREAK
0784 2A 07             BPL   R D2
                *
0786 B1 0702           CMP A E SEC    TIME CHANGED?
0789 27 F6             BEQ   R D1     LØØP TILL IT DØES
078B 20 D6             BRA   R DEMØ   PRINT TIME AGAIN
                *
078D CE 0763    D2     LDX   I DEMØ   TRANSFER DEMØ STARTING
0790 FF A048           STX   E $A048    ADDRESS TØ STACK
0793 7F A043           CLR   E $A043    CLEAR INTERRUPT MASK
0796 7E E0E3           JMP   E $E0E3  TØ 'CØNTRØL'
                *   BCD CØNVERSIØN
0799 4F         ØUT    CLR A          A HØLDS HIGH ØRDER DIGIT
079A C1 0A      Ø1     CMP B # 10     B>9 ?
079C 2B 06             BMI   R Ø2     DØNE IF NØT
079E 8B 10             ADD A # $10    A=A+$10
07A0 C0 0A             SUB B # 10     B=B-10
07A2 20 F6             BRA   R Ø1     LØØP
07A4 1B         Ø2     ABA            ØNLY WØRKS FØR NØS <80
                *
07A5 36                PSH A               VALUE TØ STACK
07A6 30                TSX              X PØINTS TØ VALUE
07A7 BD E0CA           JSR   E $E0CA  'ØUT2HS'...TØ PRINT VALUE + SPACE
07AA 31                INS              CLEAN UP STACK
07AB 39                RTS
                *
                *
                *  SET INTERRUPT REQUEST PIVØT IN MIKBUG
                *
A000                   ØRG   A $A000
A000 0734              FDB   A IRQ
                *
                *   CLEAR INTERRUPT MASK
                *    .......MUST ALWAYS REMEMBER TØ DØ THIS BEFØRE
                *     LEAVING MIKBUG, ØR THE CLØCK MAY BE STØPPED
                *     BY MIKBUG'S  'RTI'.
A043                   ØRG   A $A043
A043 00                FCB   A 0
                *
                *   SET CLØCK SET ADDRESS
A048                   ØRG   A $A048
A048 0704              FDB   A CLØCK
                *
A04A                   END   A

SYMBØL TABLE
HR     0700    MIN    0701    SEC    0702    SPLIT 0703    CLØCK 0704
IN2    0725    IRQ    0734    RTI    0762    DEMØ  0763    D1    0781
D2     078D    ØUT    0799    Ø1     079A    Ø2    07A4    ØLØAD E00A
ØBADD  E047    ØØUTL  E067    ØØUTR  E06B    ØPDAT E07E    ØØU2  E08F
ØØU2S  E0CA    ØØU4S  E0C8    ØØUS   E0CC    ØINEE E1AC    ØØUTE E1D1
```

be at the top end of the 2 K supplied with the SwTPC machine, as it was originally sold before 4 K became standard. Modifications to load at other locations are only minor. The package consists of a number of separate segments as follows.

The Clock Counters

Four bytes are reserved for the hours, minutes, seconds and deciseconds clock counters. Each actually contains the natural complement of the value, eg: (24-hours) or (60-minutes).

The Clock Initialization Routine.

On entering this routine, two pairs of two digits must be entered. These are read using the MIKBUG INHEX routine at hexadecimal E0AA. The first of the pair is multiplied by 10, by shifting and adding, and added to the second. The first sum is used to set the hours, the second, minutes. Seconds and split seconds are set to zero. The fourth digit is entered only on the time signal, generated by WWV, for example. To prevent the time from being changed while waiting for the time signal, the clock is inhibited by the SEI instruction and freed when the clock is set by the CLI instruction.

The Clock Counting Routine

This routine is entered when the processor is interrupted, provided the entry point is placed in the MIKBUG programmable memory at hexadecimal A000 and A001. The routine first checks that the clock PIA did in fact cause the interrupt, and acknowledges it by reading the PIA data register. The decisecond counter is decremented and tested. If it has not reached zero, the routine returns to the interrupted program, via the RTI. If it has, the counter *is* reset to 10 and the minutes counter decremented and tested, and so on. While it may seem that the computer has a lot to do to keep up with the clock, it utilizes only a tenth of a percent of real time during the worst case midnight rollover, and about half that normally.

A Clock Demonstration Routine

A clock demonstration routine has been included in the package and need only be loaded for testing purposes, as ordinarily the clock would be accessed by the main program. The routine prints the time, in hours, minutes and seconds at 1 or 2 second intervals depending on the printer speed. A short routine is used to convert the binary complement of the clock to binary coded decimal (BCD) so that it can be printed in hexadecimal by the OUT2HS routine in MIKBUG at hexadecimal E0CA. After sending CR/LF/* using the MIKBUG PDATA1 to send the MIKBUG MCL string, the seconds timer is read. The program waits in a loop comparing this value with the seconds timer and when a difference is found, the routine loops to print the time again. Also in the loop the routine tests the high order bit of the A data register of the control interface. If this is not 1, it means the operator pressed a key, so the routine

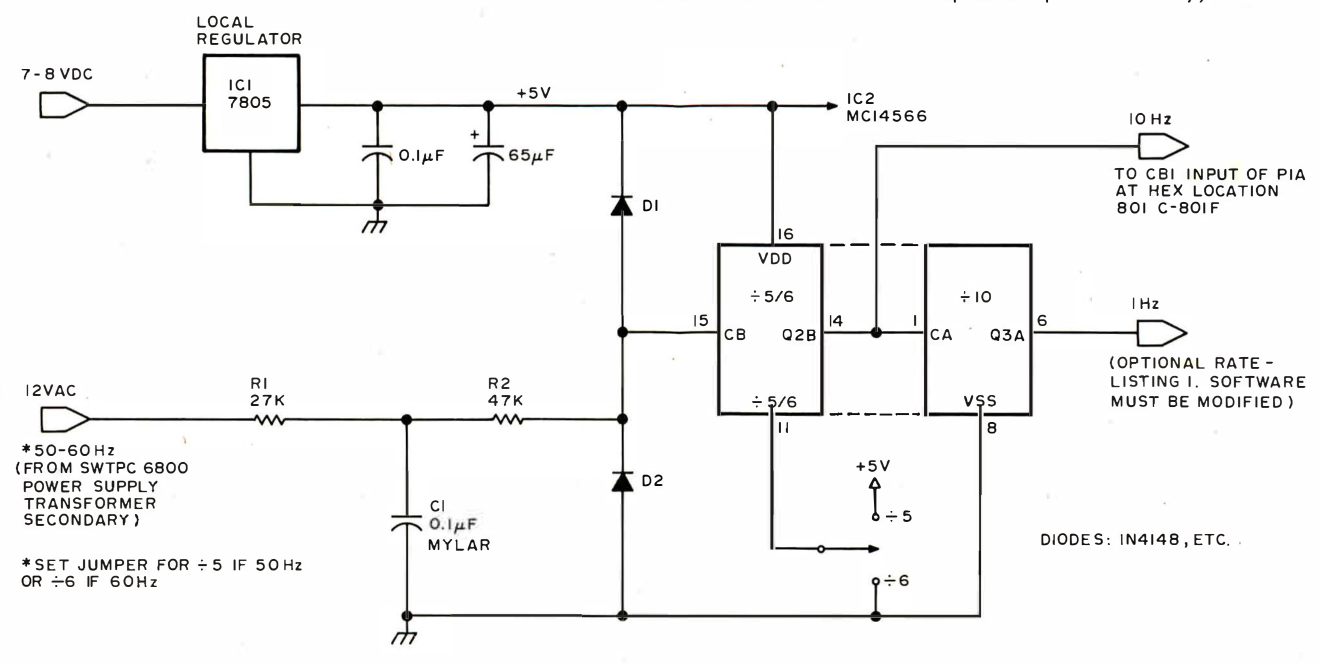


Figure 1: A way to derive the power line base of 60 cycles per second (North America) or 50 cycles per second (Europe). The low voltage secondary of the transformer in the power supply drives the Motorola MC14566, a programmable divider with ratios of 5 or 6. A second stage can optionally create 1 Hz as well as the 10 Hz signal assumed by the software of listings 1.

branches back to the MIKBUG CONTROL. This kind of approach should always be used to return to MIKBUG, for the RESET button will stop the clock by setting the interrupt mask. Also if bit 4 of condition codes on the stack (hexadecimal A043) is 1, the mask will be set upon execution of the G command, which should be avoided.

The timing pulses themselves are derived from the 50 or 60 Hz line using the circuit given in figure 1. The components can be mounted on a small piece of Micro-Vector board, supported at right angles to the base plate of the SwTPC 6800's box, near the +12 V supply board. Three short wires can then be run to one of the 12 VAC transformer leads, to the unregulated 7-8 VDC, and to ground. The output pulses can be strung directly to the C1 pin of a PIA board. The heart of the circuit is the Motorola MC14566 Industrial Time Base Generator. This MOS device contains a divide by 10 ripple counter and a divide by 5 or divide by 6 ripple counter for counting from a 50 or 60 Hz line. Pulse shapers on the inputs accept slow rise time inputs, but it is necessary to filter the line signal with R1 and C1 to remove noise. The two diodes and R2 convert the signal approximately to a square wave for the counters. ÷6 is achieved by strapping pin 11 to ground, to +5 V for ÷5.

Programming Considerations

A potentially dangerous way of moving a string of bytes from one location to another is to use the stack pointer as an index register. It is only dangerous in a case where interrupts are continuously allowed, as with this clock. For example, one might use the routine of listing 2 to move the 100 bytes starting at OLD to 100 bytes starting at NEW.

If an interrupt occurs during the execution of this segment, those bytes just before the stack pointer will be zapped with the register information, which is probably undesirable! In general, when such a technique is used to coordinate multibyte operations, it would be desirable to inhibit the interrupt. This can be done with the instruction SEI which sets the interrupt mask, thereby preventing the interrupts. The companion instruction CLI clears the mask, enabling the interrupt. Thus the segment given would be preceeded by an SEI and followed by a CLI. All is fine, provided we do not set the mask for so long that the next interrupt is lost. This is a perfect example of why at least two *full* index registers should be incorporated in each microprocessor's design. With the routine given, one can

```
OLD     RMB     100
NEW     RMB     100
SAVSP   FDB
         .
         .
         .
        STS     SAVSP         Save stack pointer
        LDS     #(OLD-1)      First byte pulled will be at OLD
        LDX     #NEW          First byte deposited at NEW
LOOP    PUL A                 Get byte
        STA A X               Store
        INX                   Advance X
        CPX     #(NEW+100)    (NEW+99) was last to move
        BNE     LOOP          Loop till done
        LDS     SAVSP         Reload stack pointer
         .
         .
         .
```

Listing 2.

move about 4 K bytes in 1/10 second, which is probably adequate for most purposes. When used with other software, you'll thus need to check carefully to make sure that any such manipulations of the stack pointer are consistent with the existence of a steady interrupt source. But once you've got a steady clock program going, a number of new possibilities will be open: time tagging files, extending the counters to keep track of days, weeks and years for scheduling personal events to be signalled when the time is ripe, etc.■

Technical Forum

Comments on Live Board Removal and Insertion

S A Stough
24802 Olive Tree Ln
Los Altos CA 94022

In reference to the Technical Forum item in the July 1977 BYTE on page 150, "Is this a valid hot board placement procedure?" I have the following comments:

The suggested procedure in the reference has a near certain probability of causing catastrophic overstress of parts because of the virtual impossibility of maintaining sufficient alignment accuracy during board removal or installation.

The close intercard spacing on most systems makes it very difficult to visually check alignment during installation. Even a rigid card frame will allow sufficient misalignment in the plane of the board to cause a momentary short between two or more pins.

Once installed, the board misalignment can typically be as much as one third of the pin to pin spacing. This reduces the margin for error to a misalignment angle of only about ten degrees. Even with card ejectors working against a rigid metal card frame, a momentary misalignment approaching this magnitude is still possible.

The procedure could be especially disastrous with the most popular 100 pin bus where logic signals and high voltage supply lines use adjacent pins.

Certain boards in high priced integrated systems use special buffering to allow power-on removal and installation where system down time is especially costly. Such would not seem to be true of personal computer systems. It would seem to be an unacceptable risk to use any live board removal or installation procedure considering the large number of damaged, degraded, and potentially degraded parts that would have to be replaced should a momentary pin to pin short occur. The parts replacement time and effort would far outweigh the time saved in not reinitializing the system for each board replacement.■

NIMBLE: The Ultimate NIM?

Irwin Doliner
POB 290
Owings Mills MD 21117

NIM is a 2 player game in which the players alternately remove counters from a pile according to some rule. The player removing the last counter is either the winner or loser depending upon the variation. One important characteristic of NIM is that exactly one player has a winning strategy available to him at the start of the game. That is to say, if the game is played "perfectly," the winner will be determined before the game begins.

Two examples will better illustrate these points. Suppose that there are 100 counters and that each player in his turn must take at least one, but no more than ten, counters. In case 1 consider the player taking the last counter as the winner and in case 2 the loser.

The winning strategy in case 1 belongs to player 1. He must take one counter on his first move, and in each successive turn take enough counters so that both players together will remove 11 counters. After player 1 has taken his turns the number of counters remaining will be 99, 88, 77, . . . , 11, and finally 0.

Player 2 has the winning strategy in case 2. In each successive turn he must take enough counters so that both players together will remove 11 counters. After player 2 has taken his turns, the number of counters remaining after each round will be 89, 78, 67, and so on down to 12 and finally 1. On his last turn, player 1 will be obliged to take the last counter and therefore lose the game.

With either of these games you will doubtless beat an unsuspecting opponent several times, even if you let him play in the favored position. But eventually, even the most casual observer will notice the invariance of your line of play regardless of what he does. Once he catches on you must find yourself another victim.

NIMBLE is the extension of NIM to a game with several piles and a slightly different rule for removing counters. It, too, has a winning strategy for exactly one of the players. This strategy is slightly more difficult to explain than the earlier ones, but much more difficult to spot. In fact, trying to learn the correct line of play from watching a knowledgeable player is like trying to catch a housefly in your hand: very often you will think you have him but when you open your hand, he's gone. It is for this reason that the game described here was called NIMBLE. It is similar to NIM but requires greater mental agility.

The rules of NIMBLE may be stated very simply. It is played with any number of piles, each of which may contain any number of counters; these numbers are fixed at the start of each game. Each player in turn

Game	Pile 1	Pile 2	Pile 3
Game 1	3	1	
Game 2	1	2	3
Game 3	9	6	4
Game 4	6	5	3

Figure 1: These are the starting positions of four simple NIM type games. If the game is played so that the player removing the last counter wins, player 1 has the advantage in game 1, and player 2 has an advantage in game 2. Player 1 should win game 3, and player 2 should win game 4 if a perfect game is played.

Jack play NIMBLE
Jack be quick,
Jack must learn
The computer's trick.

removes a quantity of counters from a single pile. He must remove at least one counter, but he may remove the entire pile. The player removing the last counter wins the game. For illustration, consider games 1 and 2 in figure 1. Player 1 has the favored position in game 1. If he is to win, he must remove two counters from pile 1 at his first play. Then player 2 must remove one of the two remaining counters and player 1 takes the other. In game 2 the advantage is not so obvious, but it belongs to player 2. No matter what player 1 does, player 2 will reduce the game to two equal piles. Then player 2 must remove from one pile whatever player 1 removed from the other.

It is not always so obvious which player has the advantage or, in fact, how to use it; for example, consider the two slightly more difficult examples in games 3 and 4. Unless you know the game strategy, it is not obvious that player 1 should win game 3, and player 2 should win game 4, assuming they play correctly. Before reading on, assume that you are player 1 in game 3 and make your first play. Did you say "Take one from pile 3?" You lose! In fact, you lose unless you said "Take 7 from pile 1." This tactic will become obvious once the winning strategy is explained.

After becoming an expert at NIMBLE, you will get greater enjoyment from the game if you can empathize with your uninitiated friends' feelings of frustration, feelings which can better be appreciated if you have been in the same situation yourself. So, if you enjoy making your own discoveries, put this program on your computer (without going too deeply into the logic) and play against it for a while as a novice.

Before typing in the program, there are some statements that may have to be changed to make them more digestible to your computer.

1. The BASIC package I used does not have a RANDOMIZE statement. Statements 300 to 350 serve that purpose.
2. Colons (:) were used in statements 10 to 70 to signify remarks in place of REM.

Listing 1: A BASIC language source listing for NIMBLE.

```
10:         *****     NIMBLE     *****
20:
30:
40:         WRITTEN BY IRWIN DOLINER
50:         AUGUST, 1976
60:
70:
80 PRINT 'NEED INSTRUCTIONS';
90 GOSUB 1900
100 IF A$='N' GOTO 290
110 PRINT
120 PRINT 'IN THIS GAME OF NIMBLE TWO PLAYERS ARE CONFRONTED WITH P'
130 PRINT '(2<P<7) PILES OF OBJECTS WITH N(I) ((0<I<=P),(0<=N(I)<64))'
140 PRINT 'OBJECTS IN PILE I. EACH PLAYER IN TURN MUST SELECT ONE'
150 PRINT 'PILE AND TAKE ANY QUANTITY FROM THAT PILE FROM 1 TO ALL.'
160 PRINT 'THE PLAYER TO TAKE THE LAST OBJECT IS THE WINNER.   '
170 PRINT 'THE GAME IS BEGUN WITH A COIN TOSS-THE WINNER OF THAT TOSS'
180 PRINT 'HAS THE RIGHT TO INDICATE A PREFERENCE FOR GOING FIRST'
190 PRINT 'OR SECOND.'
200 PRINT 'YOU INDICATE YOUR MOVE BY P,Q WHERE P=THE PILE NUMBER, '
210 PRINT 'AND Q=THE QUANTITY.'
220 PRINT 'ONCE YOU LEARN THE PROPER STRATEGY YOU SHOULD BEAT THE'
230 PRINT 'MACHINE ABOUT 50% OF THE TIME-THERE IS A WINNING STRATEGY'
240 PRINT 'WHICH THE PROGRAM USES.'
250 PRINT
260 PRINT 'G O O D    L U C K ! ! ! !'
270 PRINT
280 PRINT
290 DIM G(6,6),V(6),N(6),P(6),W(2)
300 PRINT 'PICK A NUMBER';
310 INPUT X
320 PRINT 'THANK YOU!        ';
330 FOR I=1 TO X
340 T=RND
350 NEXT I
360 I9=6
370 J9=6
380 FOR I=1 TO I9
390 V(I)=2**(I-1)
400 NEXT I
410 MAT G=ZER
420 MAT N=ZER
430 MAT P=ZER
440 I9=6
450 PRINT 'SHOULD I SET UP GAME';
460 GOSUB 1900
470 IF A$='N' GOTO 590
480 PRINT 'INDICATE DIFFICULTY LEVEL(1-5)';
490 INPUT I9
500 IF I9>63 GOTO 480
510 IF I9<1 GOTO 480
520 I9=I9+1
530 N=INT(RND*(J9-2))+3
540 FOR J=1 TO N
550 N(J)=INT(RND*(2*V(I9)-1))+1
560 GOSUB 1700
570 NEXT J
580 GOTO 730
590 PRINT 'HOW MANY PILES';
600 PRINT '(3-';J9;')';
610 INPUT N
620 IF N<3 GOTO 600
630 IF N>J9 GOTO 600
640 PRINT 'HOW MANY IN PILE NO.'
650 FOR J=1 TO N
660 PRINT J;
670 INPUT N(J)
680 IF N(J)<2*V(I9) GOTO 710
690 PRINT 'SELECT NUMBERS LESS THAN ';2*V(I9)
700 GOTO 660
710 GOSUB 1700
720 NEXT J
730 S=1
740 PRINT 'I AM ABOUT TO TOSS A COIN - CALL H OR T ';
750 T1=INT(2*RND)
760 INPUT A$
770 IF A$='H' GOTO 810
780 IF A$='T' GOTO 810
790 PRINT 'DON''T BE A WISEGUY - CALL H OR T ';
800 GOTO 760
810 IF T1 =0 GOTO 840
820 C$='H'
830 GOTO 850
840 C$='T'
850 PRINT 'THE TOSS WAS ';C$
860 GOSUB 1780
870 IF C$=A$ GOTO 900
880 PRINT 'MY CHOICE - PONDER PONDER PONDER - ';
```

Listing 1, continued:

```
890 GOTO 940
900 PRINT 'YOUR CHOICE - DO YOU WANT TO GO FIRST';
910 GOSUB 1900
920 IF A$='Y' GOTO 970
930 GOTO 990
940 FOR I=1 TO I9
950 IF P(I)=1 GOTO 990
960 NEXT I
970 PRINT 'YOU GO FIRST - ';
980 GOTO 1010
990 PRINT 'I GO FIRST - ';
1000 S=(S-1)**2
1010 IF S=1 GOTO 1400
1020 FOR I=I9TO 1 STEP -1
1030 IF P(I) = 1 GOTO 1100
1040 NEXT I
1050 J=INT(N*RND)+1
1060 IF N(J)=0 GOTO 1050
1070 T=INT(N(J)*RND)+1
1080 PRINT USING 1230,J,T
1090 GOTO 1560
1100 T=V(I)
1110 FOR J=1 TO N
1120 IF G(I,J)=1 GOTO 1140
1130 NEXT J
1140 G(I,J)=0
1150 P(I)=0
1160 FOR K=1 TO I
1170 IF P(K)=0 GOTO 1210
1180 T=T+(2*G(K,J)-1)*V(K)
1190 G(K,J)=(G(K,J)-1)**2
1200 P(K)=0
1210 NEXT K
1220 PRINT USING 1230,J,T
1230 :MY MOVE IS ###,###
1240 N(J)=N(J)-T
1250 FOR J=1 TO N
1260 IF N(J)<>0 GOTO 1370
1270 NEXT J
1280 IF S=1 GOTO 1310
1290 PRINT 'I WIN !!!!!!       ';
1300 GOTO 1320
1310 PRINT 'YOU WIN !!!!!!       ';
1320 W(S+1)=W(S+1)+1
1330 PRINT 'PLAY AGAIN';
1340 GOSUB 1900
1350 IF A$='N' GOTO 1970
1360 GOTO 410
1370 IF S=1 GOTO 1000
1380 GOSUB 1800
1390 GOTO 1000
1400 PRINT 'YOUR MOVE';
1410 INPUT J,T
1420 IF J>=1 GOTO 1460
1430 PRINT 'PICK A PILE FROM I TO ';N;
1440 INPUT J
1450 GOTO 1420
1460 IF J>N GOTO 1430
1470 IF N(J)>0 GOTO 1510
1480 PRINT 'THAT PILE IS EMPTY'
1490 GOSUB 1800
1500 GOTO 1400
1510 IF T>=1 GOTO 1550
1520 PRINT 'PICK A QUANTITY FROM 1 TO ';N(J);
1530 INPUT T
1540 GOTO 1510
1550 IF T>N(J) GOTO 1520
1560 N(J)=N(J)-T
1570 T=N(J)
1580 FOR I=I9 TO 1 STEP -1
1590 IF V(I)>T GOTO 1650
1600 T=T-V(I)
1610 IF G(I,J)=1 GOTO 1680
1620 G(I,J)=1
1630 P(I)=(P(I)-1)**2
1640 GOTO 1680
1650 IF G(I,J)=0 GOTO 1680
1660 G(I,J)=0
1670 P(I)=(P(I)-1)**2
1680 NEXT I
1690 GOTO 1250
1700 T=N(J)
1710 FOR I=I9 TO 1 STEP -1
1720 IF V(I)>T GOTO 1760
1730 G(I,J)=1
1740 T=T-V(I)
1750 P(I)=(P(I)-1)**2
1760 NEXT I
1770 RETURN
1780 PRINT USING 1790,N
1790: THERE ARE ## PILES
1800 PRINT
1810 FOR X=1 TO N
1820 PRINT X;
1830 NEXT X
1840 PRINT
1850 FOR X=1 TO N
1860 PRINT N(X);
1870 NEXT X
1880 PRINT
1890 RETURN
1900 INPUT A$
1910 IF A$<>'N' GOTO 1930
1920 RETURN
1930 IF A$<>'Y' GOTO 1950
1940 RETURN
1950 PRINT 'ANSWER ONLY Y OR N';
1960 GOTO 1900
1970 PRINT USING 1980,W(1),W(2)
1980 : FINAL SCORE - ME ###   YOU ###
1990 END
```

Figure 2: Three example moves from a typical NIMBLE game. Figure 2a illustrates the board set up before the first move. Notice that all the rows have an even number of 1s. Figure 2b shows the board after the first move. The first column now has an odd number of 1s. The second player restores the binary balance by removing four counters from the third pile and leaves an even number of 1s in each column.

(2a)

Pile Number	Quantity Decimal	Quantity Binary
1	3	0 1 1
2	6	1 1 0
3	5	1 0 1

(2b)

Pile Number	Quantity Decimal	Quantity Binary
1	3	0 1 1
2	2	0 1 0
3	5	1 0 1

(2c)

Pile Number	Quantity Decimal	Quantity Binary
1	3	0 1 1
2	2	0 1 0
3	1	0 0 1

3. PRINT USING uses a format statement rather than a format variable (see 1080, 1220, 1230, 1780, 1970, and 1980). This may not be acceptable to your BASIC.

Looking at the sample run, you will see that you have a choice of setting up the game yourself and determining the number of piles and the quantity in each pile, or letting the computer do it. If the computer sets up the game, you must select a difficulty level which determines the maximum number in each pile.

When the game is set up, the computer will simulate a coin toss; the winner will have the privilege of determining who should play first. (If the computer ever wins the toss and loses the game, look for an error in copying the program.) Once you learn the strategy, the game will be decided on the coin toss (unless you win the toss and make an error). You should begin your play by letting the computer set up the games at difficulty levels 1 and 2. When you think that you have discovered the winning strategy, test your theory at the higher levels.

If you discover the strategy for beating

→ A CALL FOR SPEAKERS ←



You read about the FIRST West Coast Computer Faire
in
Byte, Interface Age, Kilobaud, Personal Computing, etc.
held in San Francisco last April
▪ 13,000 People ▪ 200 Exhibitors ▪ 100 Speakers
over 320 pages of published *Conference Proceedings.*
Well. . .

WE'RE DOIN' IT AGAIN

The SECOND West Coast Computer Faire

will be held in
The Brand New **Convention Center** in **San Jose**
in the
middle of "Silicon Valley" – the south end of the San Francisco Peninsula
expecting
▪ 10,000-15,000 People ▪ 50-100 Speakers ▪ 150-190 Exhibitors

March 3 - 4 - 5 1978

9am-6pm *9am-6pm* *Noon-5pm*

(That's right after Compcon concludes in San Francisco)

YOU Can Be A Part Of It:

▪Talk about your latest project
Write *now* for speaker's instructions
Conference talks will be published

▪Exhibit homebrewed system
Prizes for best "homecooking"
(just like an old county fair)

▪Organize & chair Conference Section
Help gather speakers *you* want to hear
Assure the Conference has topics that interest *you*

Talks to be included in the published **Conference Proceedings** *must arrive by January 2, 1978, in the required format.*

Some of the **Conference Sections** being planned:

- Tutorials for computer novices
- Speech synthesis & speech recognition
- Computer-driven & computer-assisted music systems
- Computer graphics & video art
- Personal computers for the physically disabled
- Manufacturer tutorials on explicit systems
- Personal computers for education
- Business systems using "home" computers
- Computers & amateur radio
- Hardware & software design & implementation
- Standards for hardware, interfaces & software
- Workshops for club leaders, retailers, NL editors, etc.

Quick! **Write for more details:**
Computer Faire, Box 1579, Palo Alto CA 94302

SPEAKERS' PAPERS' DEADLINE: JAN. 2

Circle 159 on inquiry card.

N

N(1) N(2) N(3) N(4) N(5) N(6)

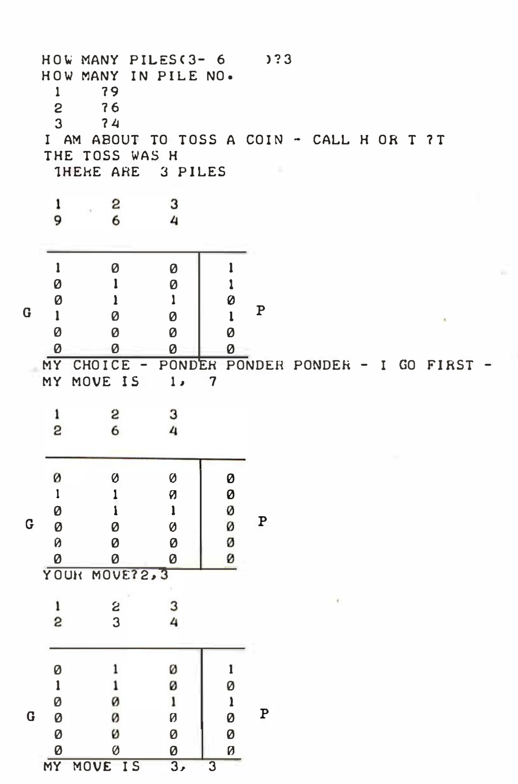
V 2^0 2^1 2^2 2^3 2^4 2^5

G

P

Figure 3: These tables represent the manner in which the information for this game is stored in memory. The matrices are defined as follows:

- V *The binary value vector* $V(I) = 2^{I-1}$.
- N *The pile quantity vector.* $N(J)$ *= the number of counters in pile J.*
- G *The binary value matrix.* $G(I, J) = 1$ *or 0 depending on whether or not* $V(I)$ *is in the binary representation of* $N(J)$.
- P *The parity vector.* $P(I) = 0$ *if row i of the G matrix has an even number of 1s; otherwise* $P(I) = 1$.



```
HOW MANY PILES(3- 6      )?3
HOW MANY IN PILE NO.
 1     ?9
 2     ?6
 3     ?4
I AM ABOUT TO TOSS A COIN - CALL H OR T ?T
THE TOSS WAS H
 THERE ARE  3 PILES

 1      2      3
 9      6      4

   ---------------------------
   1     Ø     Ø   |   1
   Ø     1     Ø   |   1
   Ø     1     1   |   Ø
 G 1     Ø     Ø   |   1     P
   Ø     Ø     Ø   |   Ø
   Ø     Ø     Ø   |   Ø
   ---------------------------
MY CHOICE - PONDER PONDER PONDER - I GO FIRST -
MY MOVE IS    1,   7

 1      2      3
 2      6      4

   ---------------------------
   Ø     Ø     Ø   |   Ø
   1     1     Ø   |   Ø
   Ø     1     1   |   Ø
 G Ø     Ø     Ø   |   Ø     P
   Ø     Ø     Ø   |   Ø
   Ø     Ø     Ø   |   Ø
   ---------------------------
YOUR MOVE?2,3

 1      2      3
 2      3      4

   ---------------------------
   Ø     1     Ø   |   1
   1     1     Ø   |   Ø
   Ø     Ø     1   |   1
 G Ø     Ø     Ø   |   Ø     P
   Ø     Ø     Ø   |   Ø
   Ø     Ø     Ø   |   Ø
   ---------------------------
MY MOVE IS    3,   3

 1      2      3
 2      3      1
```

Figure 4: A typical NIMBLE game showing the contents of G matrix and P table for each move in the game.

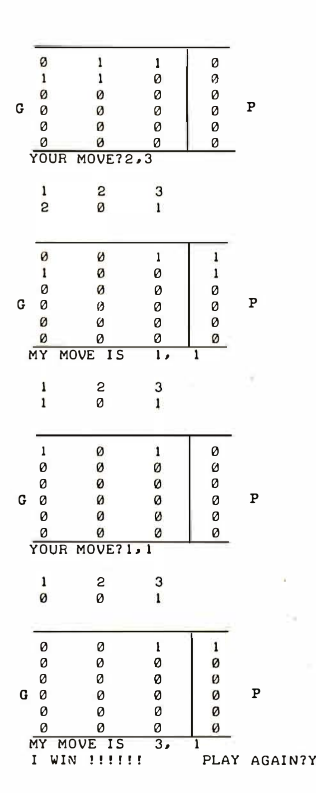
the computer, short of doctoring the program: congratulations! If not, you have probably come to the conclusion, after watching the computer's strategy for a while, that it attempts to maintain a certain kind of balance. That is absolutely correct; though not in the way you probably think.

If we lived in a binary world, this game would be very uninteresting because the strategy would be too obvious. But unfortunately man learned to count on his fingers and not his ears. We just normally think in decimal.

The following demonstration should clarify the strategy. Suppose we have 3 piles with 3, 6 and 5 in piles 1, 2 and 3 respectively. If we represent these quantities in binary, the system would look like figure 2a.

There is an even number of 1s in each column; no matter what player 1 does, this statement will no longer be true after his move. Suppose that player 1 removes 4 from pile 2: the system now looks like figure 2b. Player 2 can now restore binary balance by removing 4 from pile 3, leaving a system

Figure 4, continued:



```
   0     1     1  |  0
   1     1     0  |  0
   0     0     0  |  0
G  0     0     0  |  0    P
   0     0     0  |  0
   0     0     0  |  0
 YOUR MOVE?2,3

   1     2     3
   2     0     1

   0     0     1  |  1
   1     0     0  |  1
   0     0     0  |  0
G  0     0     0  |  0    P
   0     0     0  |  0
   0     0     0  |  0
 MY  MOVE  IS     1,  1

   1     2     3
   1     0     1

   1     0     1  |  0
   0     0     0  |  0
   0     0     0  |  0
G  0     0     0  |  0    P
   0     0     0  |  0
   0     0     0  |  0
 YOUR MOVE?1,1

   1     2     3
   0     0     1

   0     0     1  |  1
   0     0     0  |  0
   0     0     0  |  0
G  0     0     0  |  0    P
   0     0     0  |  0
   0     0     0  |  0
 MY  MOVE  IS     3,  1
 I WIN !!!!!!           PLAY AGAIN?Y
```

that looks like figure 2c. The game will continue this way, with player 1 disrupting the balance and player 2 restoring it. Player 1 must ultimately leave a single pile which player 2 will remove. Therefore, if the initial system is in binary balance it is preferable to go second; otherwise, it is preferable to go first. The strategy is simply to restore binary balance each time your opponent disrupts it.

Now that you know the winning strategy, you will want to learn how the program knows to play it. First we must think of writing binary numbers vertically from bottom up rather than horizontally from left to right. For example, the numbers 3, 5, 7 and 8 would be represented as:

3	5	7	8
1	1	1	0
1	0	1	0
0	1	1	0
0	0	0	1

This method is used to represent the pile quantities in the G table (matrix).

Figure 3 demonstrates the relationships between the major program tables and how they are used to find the optimal move.

When it is the computer's turn to play, it looks at the P vector, statements 1020 to 1040 of listing 1. If it is not equal to 0 there is no optimal move and the computer plays at random as shown in statements 1050 to

Figure 5: A typical NIMBLE game series.

```
RUN                                SAMPLE RUN

NIMBLE       20:16    08/31/76

NEED INSTRUCTIONS?Y

IN THIS GAME OF NIMBLE TWO PLAYERS ARE CONFRONTED WITH P
(2<P<7) PILES OF OBJECTS WITH N(I) ((0<I<=P),(0<=N(I)<64))
OBJECTS IN PILE I. EACH PLAYER IN TURN MUST SELECT ONE
PILE AND TAKE ANY QUANTITY FROM THAT PILE FROM 1 TO ALL.
THE PLAYER TO TAKE THE LAST OBJECT IS THE WINNER.
THE GAME IS BEGUN WITH A COIN TOSS-THE WINNER OF THAT TOSS
HAS THE RIGHT TO INDICATE A PREFERENCE FOR GOING FIRST
OR SECOND.
YOU INDICATE YOUR MOVE BY P,Q WHERE P=THE PILE NUMBER,
AND Q=THE QUANTITY.
ONCE YOU LEARN THE PROPER STRATEGY YOU SHOULD BEAT THE
MACHINE ABOUT 50% OF THE TIME-THERE IS A WINNING STRATEGY
WHICH THE PROGRAM USES.

G O O D     L U C K ! ! ! !


PICK A NUMBER?74
THANK YOU!         SHOULD I SET UP GAME?Y
INDICATE DIFFICULTY LEVEL(1-5)?1
I AM ABOUT TO TOSS A COIN - CALL H OR T ?H
THE TOSS WAS T
 THERE ARE   5 PILES

 1      2      3      4      5
 2      1      3      1      3
MY CHOICE - PONDER PONDER PONDER - I GO FIRST -
MY MOVE IS    1,  2

 1      2      3      4      5
 0      1      3      1      3
YOUR MOVE?5,3
MY MOVE IS    3,  3

 1      2      3      4      5
 0      1      0      1      0
YOUR MOVE?2,1
MY MOVE IS    4,  1
I WIN !!!!!!       PLAY AGAIN?Y
SHOULD I SET UP GAME?Y
INDICATE DIFFICULTY LEVEL(1-5)?5
I AM ABOUT TO TOSS A COIN - CALL H OR T ?H
THE TOSS WAS T
 THERE ARE   3 PILES

 1      2      3
 40     55     60
MY CHOICE - PONDER PONDER PONDER - I GO FIRST -
MY MOVE IS    1, 29

 1      2      3
 11     55     60
YOUR MOVE?3,10
MY MOVE IS    1,  6

 1      2      3
 5      55     50
YOUR MOVE?2,37
MY MOVE IS    3, 27

 1      2      3
 5      18     23
YOUR MOVE?3,22
MY MOVE IS    2, 14

 1      2      3
 5      4      1
YOUR MOVE?1,2
MY MOVE IS    2,  2

 1      2      3
 3      2      1
YOUR MOVE?1,1
MY MOVE IS    3,  1

 1      2      3
 2      2      0
YOUR MOVE?1,1
MY MOVE IS    2,  1

 1      2      3
 1      1      0
YOUR MOVE?1,1
MY MOVE IS    2,  1
I WIN !!!!!!       PLAY AGAIN?Y
SHOULD I SET UP GAME?N
HOW MANY PILES(3- 6     )?3
HOW MANY IN PILE NO.
 1      ?9
 2      ?6
 3      ?4
I AM ABOUT TO TOSS A COIN - CALL H OR T ?T
THE TOSS WAS H
 THERE ARE   3 PILES

 1      2      3
 9      6      4
MY CHOICE - PONDER PONDER PONDER - I GO FIRST -
MY MOVE IS    1,  7

 1      2      3
 2      6      4
```

Figure 5, continued:

```
YOUR MOVE?2,3
MY MOVE IS    3,  3

 1      2      3
 2      3      1
YOUR MOVE?2,3
MY MOVE IS    1,  1

 1      2      3
 1      0      1
YOUR MOVE?2,1
THAT PILE IS EMPTY

 1      2      3
 1      0      1
YOUR MOVE?1,1
MY MOVE IS    3,  1
I WIN !!!!!!        PLAY AGAIN?Y
SHOULD I SET UP GAME?N
HOW MANY PILES(3- 6      )?3
HOW MANY IN PILE NO.
 1      ?6
 2      ?5
 3      ?3
I AM ABOUT TO TOSS A COIN - CALL H OR T ?H
THE TOSS WAS T
 THERE ARE  3 PILES

 1      2      3
 6      5      3
MY CHOICE - PONDER PONDER PONDER - YOU GO FIRST - YOUR MOVE?1,2
MY MOVE IS    3,  2

 1      2      3
 4      5      1
YOUR MOVE?2,2
MY MOVE IS    1,  2

 1      2      3
 2      3      1
YOUR MOVE?2,3
MY MOVE IS    1,  1

 1      2      3
 1      0      1
YOUR MOVE?1,1
MY MOVE IS    3,  1
I WIN !!!!!!        PLAY AGAIN?N
 FINAL SCORE - ME    4    YOU    0
```

1080. If P equals 0 the computer determines the largest I for which P(I) = 1 and then selects the smallest j for which G (I,J) = 1, ie: it picks the first pile which contains V(I). The computer then concentrates on column J and takes the 1's complement of G(I,J) whenever P(I) = 1, statements 1100 to 1220. Figure 4 demonstrates this process by showing the contents of tables G and P after each move in the playing of game 3. If you work this through carefully, you will see how the computer uses these tables to find an optimal move when one is possible. Figure 5 shows a sample run of NIMBLE.

Even though you now know the secret of NIMBLE, you can still have fun with the game. You can test your ability to rapidly convert numbers to binary at difficulty levels 4 and 5. You can make the game more difficult by increasing the sizes of the arrays (also I9 and J9) to allow larger values and more piles; or you can merely impress your friends by beating the computer when they can't. However you use NIMBLE, I hope that you have as much fun playing it as I had writing it.■

Circle 140 on inquiry card.

Continued from page 78

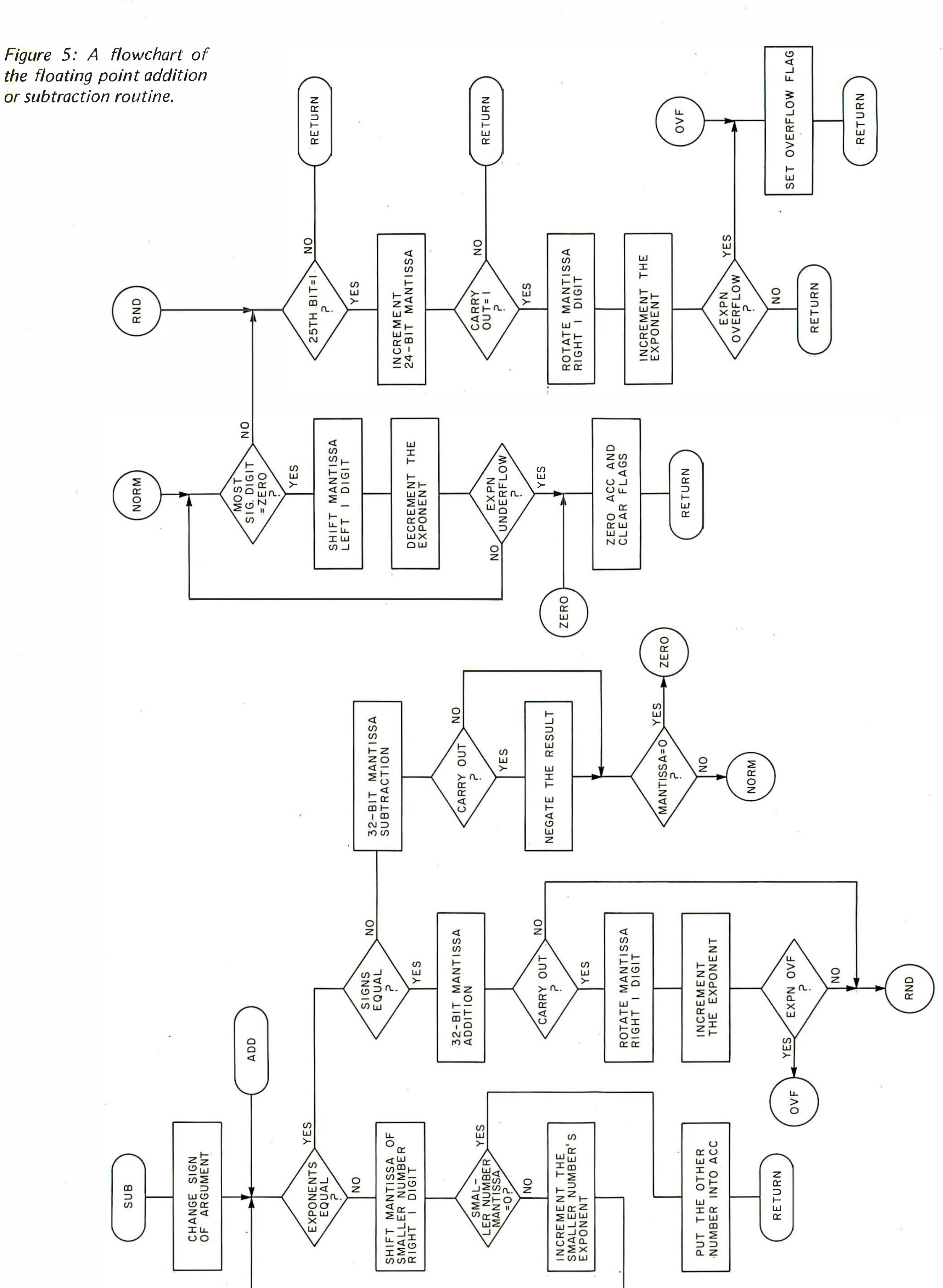
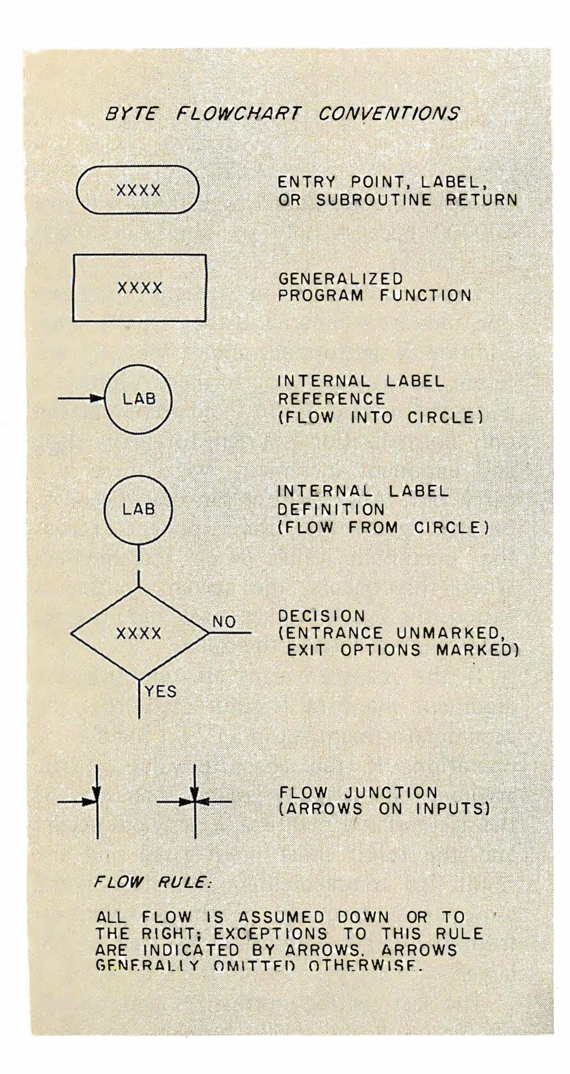


Figure 5: A flowchart of the floating point addition or subtraction routine.



number of digits can be expanded.

This format has its list of disadvantages, though; but for these the commercial computer industry might have adopted it long ago. The program size required for performing just the basic operations and the conversion routines is about the same as for the other formats, but execution times are significantly slower. Many hobbyists are not as concerned with the number of milliseconds as with the number of bytes, but another disadvantage is the larger memory required to store the floating point numbers. For most assembly language applications the impact is negligible. It does become noticeable, however, when the floating point package is part of higher level language programs such as interpreters or compilers. One major disadvantage is more subtle. Many of the transcendental functions are best implemented using algorithms which are binary based. Using these algorithms, the BCD format is awkward at best and at worst consumes large quantities of time and memory.

The binary floating point format provides the fastest execution times, despite the fact that its format allows representation of 7 digit numbers at all times. Because the entire format is in binary, implementing the basic operations and all of the transcendental functions is easier than when using either of the other two formats.

The major drawback is the small range of numbers representable, relative to the other formats (10^{+38} to 10^{-38}). This is because its exponent is only a power of two compared with bases of 10 and 16 respectively. Two other minor drawbacks are the need for routines to convert floating point numbers from a decimal base to a binary base (and vice versa), and the need to expand the binary format to perform actual calculations.

The hexadecimal floating point format permits a much larger number range (10^{+76} to 10^{-76}) than the binary format, and the conversion routines are similar for both. Although slightly slower than the binary format, the hexadecimal format is still much faster than any BCD format of comparable capability.

It is somewhat more difficult to implement scientific functions such as square root, exponential and logarithm with this format than with the binary format, and its precision is not as great as the binary format's precision because it is digit rather than bit oriented. Even though the most significant digit is nonzero, the most significant three bits of the digit itself may be zeroes, resulting in only 21 bits of accuracy. This translates to only six digits of accuracy.

In describing the four basic floating point operations and the format conversions, the hexadecimal format will be used to illustrate examples.

Floating Point Operations

The software uses three floating point registers, an accumulator, argument register and scratch register. The floating point accumulator contains one of the operands prior to a calculation, and the result after the calculation is performed. The argument register contains the other operand, which is loaded by the routine, and the scratch register is used to hold temporary results.

In each of the basic operations there are two parts: exponent calculation and mantissa calculation. Fixed point operations require only the mantissa calculation, which turns out to be the easier of the two.

Add and Subtract Routine

Figure 5 is a flowchart of the add and subtract routine. The two operations are described together because the algorithms

A =	.100000	X	16^1
B =	.FFFFFF	X	16^0

Figure 6a: Two numbers A and B, which differ from one another by less than one part in 2^{24}, but which were represented as two different numbers.

	Mantissa	Guard Byte		
A =	.100000	00	X	16^1
B =	.0FFFFFF	F0	X	16^1

Figure 6b: The same numbers as figure 6a, but with B shifted to the right one digit, and the extra digit stored in the guard byte in preparation for the subtraction shown in figure 6c. This shifting aligns mantissa radix points (makes exponents equal).

	Mantissa	Guard Byte						
A =	.100000	00	X	16^1				
–B =	–.0FFFFF	F0	X	16^1				
C =	.000000	10	X	16^1	=	.100000	X	16^{-5}

Figure 6c: The subtraction of B from A to give C. There is only one significant digit in the result, which is entirely located within the guard byte.

A =	.100000	X	16^1				
–B =	–.0FFFFF	X	16^1				
C =	.000001	X	16^1	=	.100000	X	16^{-4}

Figure 6d: If the guard byte is omitted, as in this example, the apparent result is off by a factor of 16 due to truncation prior to the mantissa addition (or subtraction).

are identical except for a sign change before executing a subtract.

The add and subtract routine consists of three functionally separate sections. The first prepares the numbers for the operation by aligning the radix points. This is analogous to aligning the decimal points for an addition or subtraction of decimal numbers. The addition or subtraction is then performed and the result normalized.

The radix points are aligned by shifting the mantissa of the smaller number right one digit and incrementing its exponent until the exponents are equal. When shifting right, the last eight bits shifted out are saved in the guard byte in order to maintain accuracy. During the shifting and incrementing loop, the 32 bit mantissa, including the guard byte, should be checked for all zeroes (a situation which implies that one operand is too small to affect the other). This is to avoid shifting insignificant zeroes. For example, 0.0001 added to 100000 will give 100000 because only six significant digits are retained.

In the second section the signs of the two operands are compared. If they are the same, addition is performed, and if they are different, subtraction is performed. Addition is a straightforward 32 bit fixed point add; the only normalization is a right rotate one digit and exponent increment when there is a carry out. An overflow can only occur if, on the right rotate, the exponent exceeds the maximum value when incremented. When this occurs, the current routine is exited, the overflow flag is set, and program control is returned to the caller.

If the mantissa signs are opposite, the argument mantissa is subtracted from the accumulator mantissa in a 32 bit fixed point operation. If the absolute value of the argument mantissa is greater than that of the accumulator mantissa, a carry out occurs and the result must be negated and the result sign complemented. The effect is the same as subtracting the smaller mantissa from the larger and using the sign of the larger.

The last section normalizes and rounds off the result and checks for exponent overflow and underflow. Normalization consists of shifting the mantissa digits left until the most significant digit is nonzero. For each shift, the exponent must be incremented and checked for overflow. Only 24 bits of mantissa are saved. Therefore, the 25th bit of the temporary result determines whether the mantissa is to be rounded up or not. For example, if the hexadecimal result were 10000094, it would be rounded up to 100001, whereas a result of 10000048 would not.

If the guard byte and a round off operation are not used in an addition, one bit of significance could be lost. By comparison, subtraction without a guard byte could mean a difference of an order of magnitude. Two numbers can be different by less than one part in 2^{24} and yet be represented as two different numbers (A and B in figure 6a). When one is subtracted from the other, the smaller must be shifted right in order to align the radix points. The guard byte stores the shifted out digit (figure 6b) and retains the only significant digit of the result (figure 6c). Without a guard byte the significant digit may be off by a factor of 16 (figure 6d).

IF YOU'RE NOT SUBSCRIBING TO CREATIVE COMPUTING, YOU'RE NOT GETTING THE MOST OUT OF YOUR COMPUTER.

No computer magazine gives you more applications than we do! Games. Puzzles. Sports simulations. CAI. Computer art. Artificial intelligence. Needlepoint. Music and speech synthesis. Investment analysis. You name it. We've got it. And that's just the beginning!

Whatever your access to computer power—home computer kit, mini, time-sharing terminal—*Creative Computing* is on your wavelength. Whatever your computer application—recreation, education, business, household management, even building control—*Creative Computing* speaks your language.

Read through pages of thoroughly documented programs with complete listings and sample runs. All made easy for you to use. Learn about everything from new software to microprocessors to new uses for home computers. And all in simple, understandable terms. And there's still more. *Creative Computing* discusses creative programming techniques like sort algorithms, shuffling and string manipulation to make your own programming easier and more efficient.

We can even save you time and money. Our extensive resource section is filled with all kinds of facts plus evaluations of hundreds of items. Including microcomputers, terminals, peripherals, software packages, periodicals, booklets and dealers. We also give you no-nonsense equipment profiles to help you decide which computer is best for you—before you spend money on one that isn't.

We've got fiction too. From the best authors in the field, like Asimov, Pohl and Clarke. Plus timely reviews of computer books, vendor manuals and government pamphlets. And so much more!

Isn't it about time you subscribed to *Creative Computing*? It's the smart way to get the most out of your computer.

Complete this coupon and mail it today. Or for fast response, call our toll-free hot line. (800) 631-8112. (In New Jersey call (201) 540-0445).

I want to get the most out of my computer.
Please enter my subscription to:

creative computing

Term	USA	Foreign Surface	Foreign Air
□ 1-year	□ $ 8	□ $ 12	□ $ 20
□ 2-year	□ $ 15	□ $ 23	□ $ 39
□ 3-year	□ $ 21	□ $ 33	□ $ 57
□ Lifetime	□ $300	□ $400	□ $600
□ Vol. 1 Bound	□ $ 10	□ $ 12	□ $ 15
□ Vol. 2 Bound	□ $ 10	□ $ 12	□ $ 15

□ Payment Enclosed
□ Visa/Bank Americard □ Master Charge
Card No. ____________

□ Please bill me ($1.00 billing fee will be added; foreign orders must be prepaid)

Name ____________

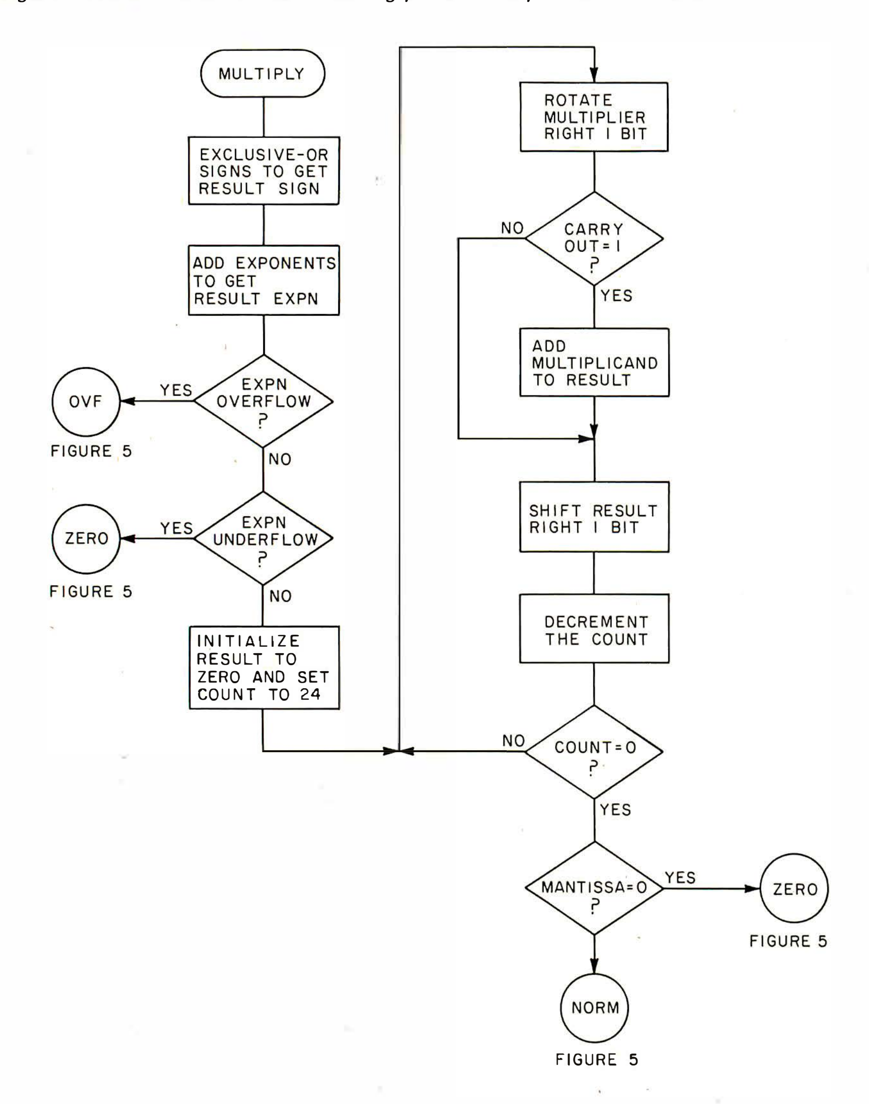
Address ____________

City ________ State ______ Zip ______

Send to: Creative Computing, Atten: Inez
P.O. Box 789-M, Morristown, N.J. 07960

Circle 39 on inquiry card.

Figure 7: A flowchart for the floating point multiplication routine.



Multiplication

Figure 7 is a flowchart of the multiplication routine. Calculation of the exponent for the multiplication and division routines is achieved by adding or subtracting the operand exponents respectively. Since the exponents are in excess-64 notation, the offset (64) will have to be subtracted from or added to the result. If the resultant exponent is less than the smallest exponent or greater than the largest, an underflow or overflow condition exists and the appropriate action is taken (for example, displaying an error message or setting the result to a fixed value). Sign calculation for both multiply and divide is a simple exclusive or of the two operand signs.

The partial product method is the most widely used in fixed point multiplications, decimal or binary based. Using binary numbers, this algorithm rotates the multiplier right one bit and tests the bit rotated out. The multiplicand is conditionally added to the accumulated result if the bit is a one. The result is then rotated right one bit, retaining 32 bits, and the whole procedure repeated for all 24 bits of the multiplier. *[An example of this algorithm implemented in hardware was found in the article "This Circuit Multiplies" by Tom Hall, page 36 in July 1977 BYTE . . .* CH*]*

Though the fixed point calculation is straightforward and uncomplicated, it is extremely time consuming because the loop is repeated 24 times. One method of reducing the execution time is to cut out all subroutines within the loop and use only in line code. A complete multiplication routine can then have a worst case multiply time of about 2.5 ms using an 8080 processor with 2 MHz clock.

Division

Figure 8 is a flowchart of the division routine. The fixed point divide algorithm is analogous to the partial product method and is also commonly used. It compares the absolute value of the divisor to that of the dividend. If it is equal to or less than the dividend's absolute value, it is subtracted from the dividend, and a one is rotated into the least significant bit of the quotient. Otherwise there is no subtraction and a zero is rotated in. The dividend is then shifted left one bit and the loop repeated for a total of 32 times, generating a 32 bit quotient. Long division by hand goes through the identical procedure, but it operates on digits instead of bits.

Since more processing is done in each loop cycle than in the multiply routine, division execution times are longer than multiplication times. The worst case times are still around 5 ms for an 8080 with 2 MHz clock.

In both the multiply and divide routines, the normalization procedure is identical to the one in the subtract routine. Therefore it usually turns out to be shared code.

These routines are the core for other floating point functions such as format conversions and scientific mathematical functions. Because of this it is important that these routines execute as fast as possible so that the other functions' execution times are not increased to several seconds instead of fractions of seconds.

BCD to Binary and Binary to BCD conversions are probably the most difficult to implement in a binary floating point package. There are several simple methods of converting integers from one format to the other, but I haven't seen any published literature to date on either floating point arithmetic or number base conversions.

The methods described here were chosen because of their simplicity rather than their speed. The slow base conversions are still relatively fast compared to the character oriented input and output operations in which they are used, so for most purposes the conversion speed is not noticeable.

Decimal to Binary Conversion

The Decimal to Binary (DB) routine (figure 9) converts a free format floating point BCD number in ASCII to binary floating point format, converting from ASCII BCD floating point to formatted BCD floating point, and then to binary floating point in one operation.

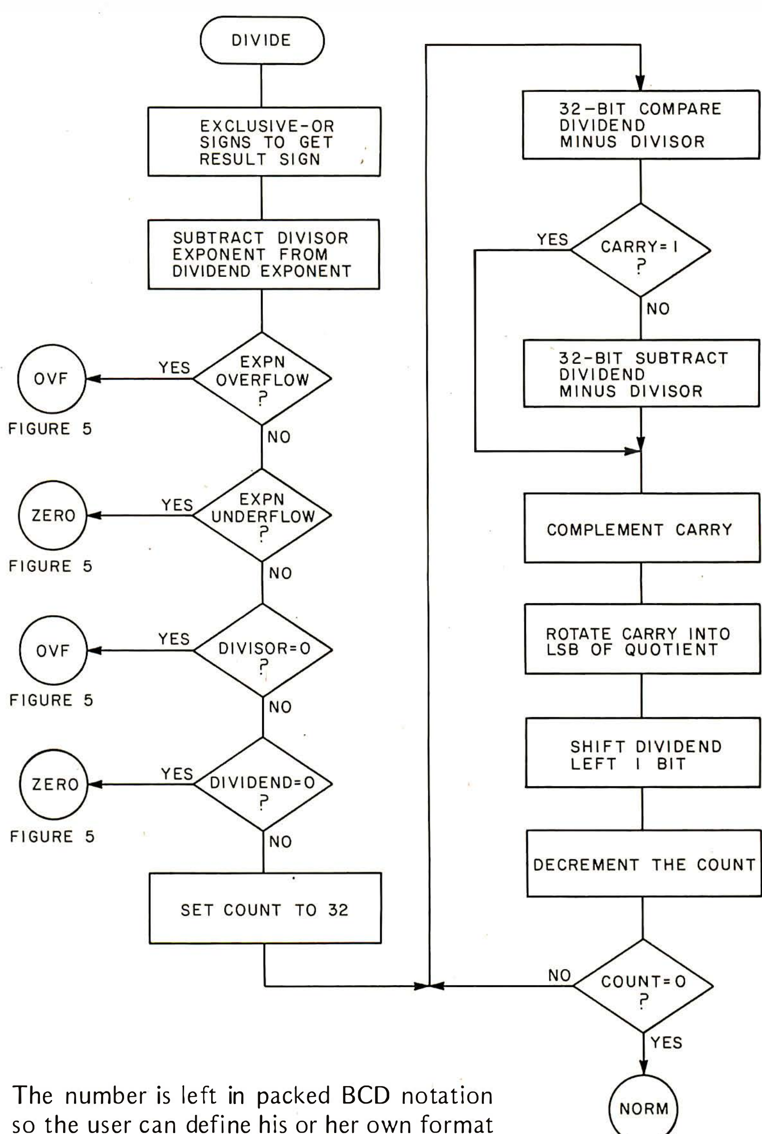
After initialization the DB routine first checks for a plus or minus sign, which is optional. It ignores a plus sign and sets a flag if there is a minus sign. It then reads in one or more digits (and possibly a decimal point). When it encounters a decimal point, it tests a flag to see if another decimal point has already occurred and sets the flag if not. If a decimal point has already occurred, the routine jumps to the last section. For each decimal digit input, the routine multiplies the accumulated result by ten in floating point format, creates a floating point number from the digit, and adds the number to the accumulated result. If a decimal point has previously occurred, a decimal exponent count is decremented, keeping track of the number of digits in the fractional part. This process is repeated until a character which is neither a digit nor decimal point has occurred, at which point control passes on to the exponent evaluation routine.

Here the decimal exponent of the number, if any, is processed. The routine first searches for the presence of an E character. If none is present, control jumps to the last section. If the character is present, one or two BCD digits are inputted with an optional plus or minus sign. The BCD digits are converted to an 8 bit binary, two's complement number and added to the decimal exponent count.

Finally, the mantissa is normalized by either repeatedly multiplying or dividing by ten, depending upon the decimal exponent count. Multiplication is performed if the count is greater than zero, and division is performed if it is less than zero. The count is either decremented or incremented respectively toward zero for every multiplication or division. When the count reaches zero, the sign is corrected if the number is negative, and the routine returns.

The Binary to Decimal (BD) routine shown in figure 10 converts a binary floating point number to packed BCD floating point.

Figure 8: A flowchart for the floating point division routine.



The number is left in packed BCD notation so the user can define his or her own format for the decimal point and exponent.

Initially, the binary number is normalized so that it is in the range of 0.1 to 1.0, with a decimal exponent kept separate. This is done by repeatedly multiplying or dividing by 10 until the number is equal to or greater than 1.0 and less than 10.0, and then dividing it by 10.0. During this operation, each multiplication or division by 10 is tabulated in a count. Next, a round off of 0.0000005 is added and a correction, if necessary, is made to make sure the number remains between 0.1 and 1.0.

The number is then converted to a binary fixed point fraction, and finally to a BCD fixed point fraction of eight digits, but accurate to only six digits because of the added round off.

After completing mantissa conversion, the binary count of the decimal exponent is converted to a signed BCD pair and stored with the BCD fraction.

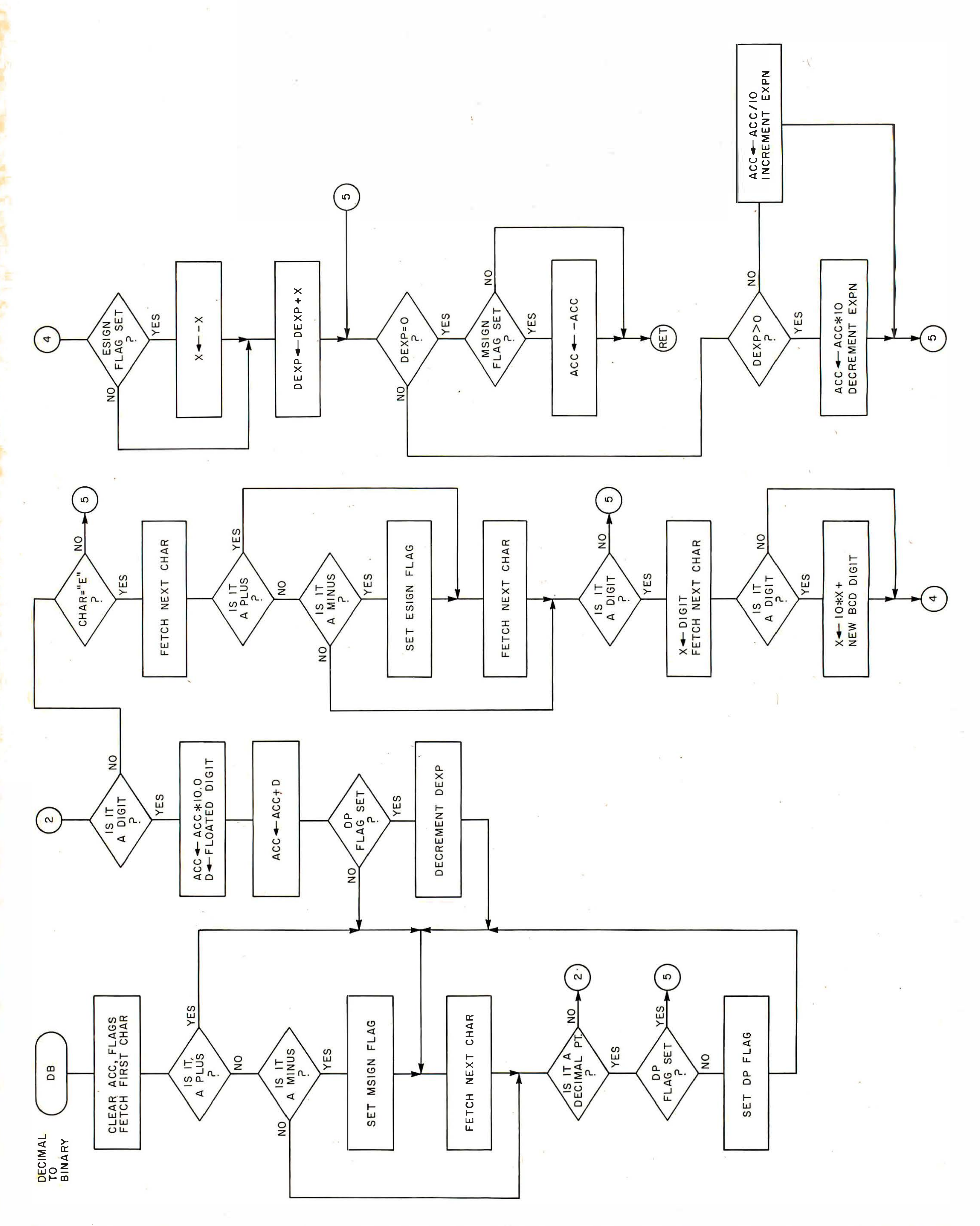


Figure 9: Flowchart of a decimal to binary routine used to convert a free format floating point BCD number in ASCII format to binary floating point format.

NOTICE to all BYTE magazine subscribers and would-be subscribers

SAVE MONEY! As of January 1, 1978, we are raising our subscription rates to meet the increased costs of producing and mailing *BYTE* to our more than 110,000 monthly readers. The new domestic subscription rates, effective January 1, 1978:

Current Rates		Rates Effective January 1, 1978	
One year U.S.	— $12	One year U.S.	— $15
Two years U.S.	— $22	Two years U.S.	— $27
Three years U.S.	— $32	Three years U.S.	— $39

If you already subscribe to *BYTE* and still have several months to go before expiration of your subscription, you can still take advantage of current rates for renewal (even up to three years). Use coupon below. When your present subscription expires, your renewal order commences.

If you are not yet a subscriber to *BYTE*, the leading magazine for the creative home computer experimenter, don't delay . . . mail this coupon and start getting your own copies of this invaluable magazine, before new rates become effective.





USE THIS MONEY-SAVING COUPON TODAY

To new subscribers:

Read your first copy of *BYTE*, if it's everything you expected, honor our invoice. If it isn't, just write "CANCEL" across the invoice and mail it back. You won't be billed, and the first issue is yours.

Allow 6 to 8 weeks for processing.

Circle 20 on inquiry card.

BYTE Subscription Dept. • P.O. Box 361 • Arlington, Mass. 02174

PLEASE ENTER MY SUBSCRIPTION FOR:

☐ One year $12 (12 issues) ☐ Two years $22 ☐ Three years $32

☐ Check enclosed (entitles you to bonus of one extra issue)

☐ Bill me ☐ Bill BankAmericard/Visa ☐ Bill Master Charge

Card Number: ☐☐☐☐☐☐☐☐☐☐☐☐☐☐ Expiration Date: ____________

Signature: ____________ Name *(please print)* ____________

Address: ____________

City: ____________ State/Country: ____________ Code: ____________

Offer expires December 31, 1977

FOREIGN RATES FOR ONE YEAR: *(Please remit in U.S. Funds)*

☐ Canada or Mexico $17.50 (Air *delivered)* ☐ Europe $25 (Air delivered)

☐ All other countries except above: $25 *(Surface delivery)*

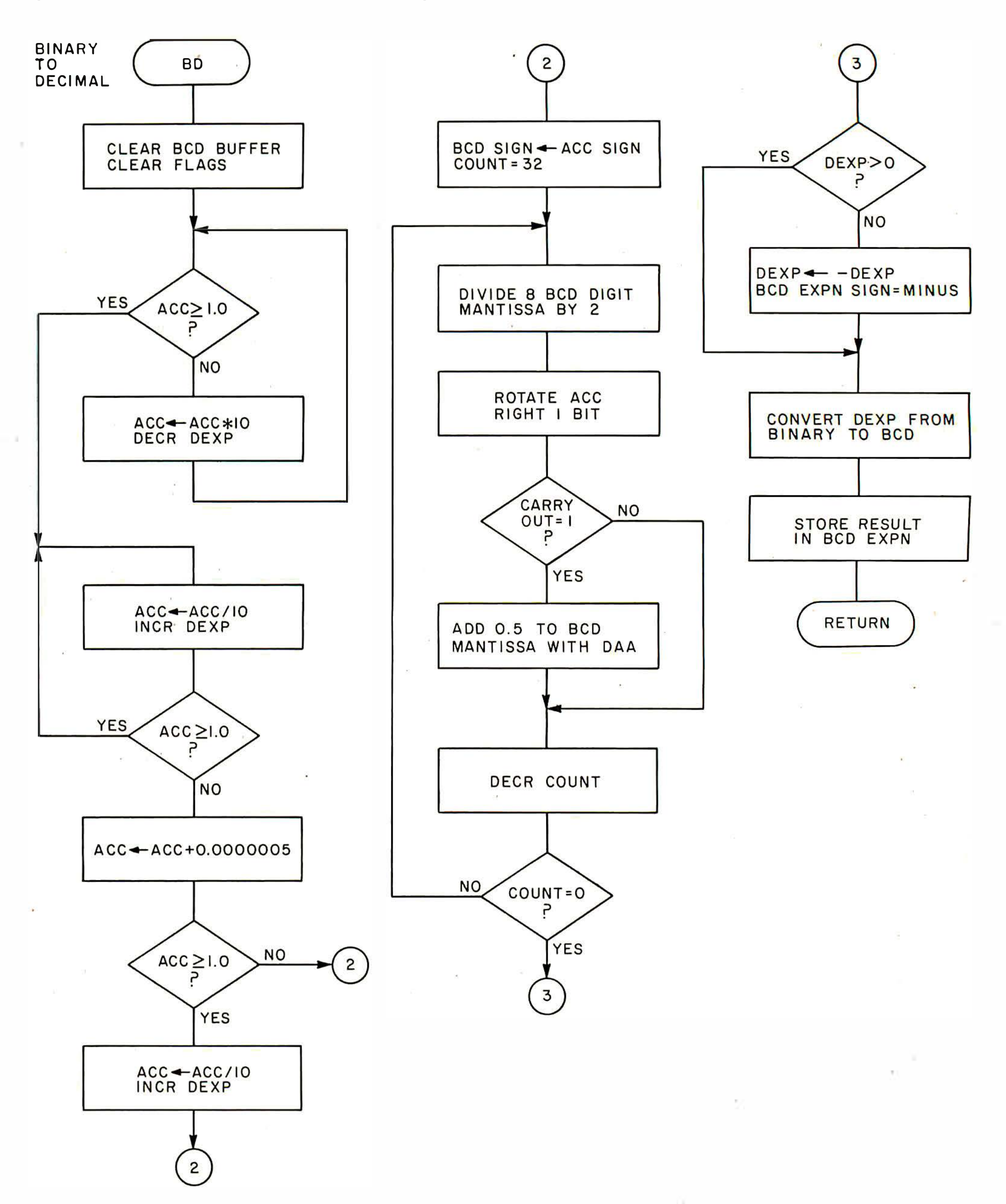
Air delivery available on request

These two algorithms for conversion of bases between BCD (base 10) and binary (base 2) are valid for any binary floating point format, not just the one used here.

Concluding Remarks

It is hoped that this discussion along with the flowchart specifications of the algorithms can be used by readers as a basis for coding a floating point arithmetic package for any general purpose microprocessor system. I have used this information in particular to code an 8080 version of the routines for the basic arithmetic functions, as well as extensions for functions such as square root, exponential, natural logarithm, sine and cosine, and arc tangent. The extensions all use the basic multiplication, division, addition and subtraction operations to evaluate the more complex functions involved. Readers interested in a detailed copy of this 8080 mathematical function software documentation can purchase it for $10 by writing to me at POB 447, Maynard MA 01754. ■

Figure 10: Flowchart of a binary to decimal conversion routine used to convert a binary floating point number to packed BCD floating point format.



SWTPC Business Programs

Math Package. 5 byte mantissa, 1 byte exponent 11½ digit accuracy. Fortran-type formatting (Integer, Floating Point and Free Formatting on both read and write) $25.00

Letter Writing Editor. No line numbers needed. Subset of Honeywell's 1648 Time Sharing Editor. $25.00

Payroll. Phone for details.

Accounts Receivable. Phone for details.

Stock Club — Marketing Evaluation. This is the same program used by Custom Computer Services to provide accounting for clubs belonging to the National Association of Investment Clubs.

Disk Based SWTPC Business Systems. Phone for details.

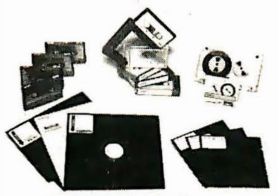
Dealer for SWTPC, Smoke Signal Broadcasting, Percom, Teletype®, GE, TI, Centronics, Motorola, etc.

To Order: Include 3% postage. Illinois residents add 5% sales tax.

AAA Chicago Computer Center
3007½ W. Waveland Avenue
Chicago, IL 60618
(312) 539-5833 (3 PM – 6 PM)

Circle 1 on inquiry card.

SUPPLIES



- FLOPPY DISKS, MINI OR STANDARD MEMOREX OR 3M
- 3M DATA CARTRIDGES DC300A, DC100A
- 3M DIGITAL CASSETTES
- 3M OR MEMOREX AUDIO CASSETTES, C-60
- 3M DISK CARTRIDGES

WE OFFER:

- COMPETITIVE PRICING
- IMMEDIATE DELIVERIES (Any Quantity)
- UNCONDITIONAL GUARANTEE

BETA BUSINESS SYSTEMS
8369 VICKERS ST., #G
SAN DIEGO, CA 92111
(714) 565-4505

Circle 11 on inquiry card.

computer depot inc.[T.M.]

3515 W. 70th Street
Minneapolis MN 55435
612-927-5601

Upper Midwest Headquarters for

PROCESSOR TECHNOLOGY, IMSAI, POLYMORPHIC SYSTEMS, DIGITAL GROUP, CROMEMCO, VECTOR GRAPHICS, TDL, SEALS, TARBELL, MICRO DESIGNS, NORTH STAR, NATIONAL SEMI-CONDUCTOR, FAIRCHILD SEMI-CONDUCTOR, MOSTEK, E & L INSTRUMENTS, SAMS, TAB, HAYDEN, WILEY, MC-GRAW HILL

Catalog Free

C40

Circle 28 on inquiry card.

COMPUTER MART OF NEW HAMPSHIRE

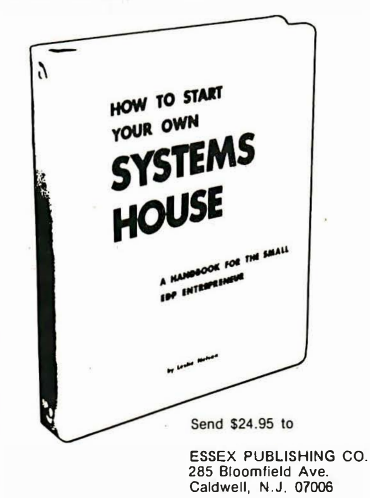
DEALERS FOR TDL ICOM IMSAI DIGITAL GROUP KIM-1

SOFTWARE INCLUDES:
APPLE II
8K BASIC EXT. BASIC
TEXT EDITOR DISK BASIC
WORD PROCESSOR
MACRO-ASSEMBLER

170 MAIN STREET NASHUA
(603) 883-2386

Circle 33 on inquiry card.

If you can program a mini and own one, you may have almost everything to start your own software house. Find out how, from this 200-page, detailed manual by an EDP professional who successfully did it.



Send $24.95 to

ESSEX PUBLISHING CO.
285 Bloomfield Ave.
Caldwell, N.J. 07006

Circle 57 on inquiry card.

SIX PORT I/O CONTROLLER

- Program Control of Ports
- S100 Buss Compatible
- Parallel I/O
- Single Printed Circuit Board

$149.00

VIDEO SYSTEM

- 30 x 64 Display — 1920 Char.
- 128 Character Buffer
- 1 μsec Read/Write Time
- Extensive Software Provided
- Programmable Display Attribute Characters

$349.00

Available in Kits or Assembled
BankAmericard and Master Charge Accepted
Dealer or Club Discounts Available

IOR
Box 28823 Dallas, Texas 75228

Circle 76 on inquiry card.



KNOWING THE TRICKS OF THE TRADE CAN SAVE YOU TIME AND MONEY DEVELOPING BUSINESS SYSTEMS.



REVEALS TRICKS NEVER BEFORE AVAILABLE TO YOU -- ABSOLUTELY GUARANTEED OR YOUR MONEY BACK. SEND $14.95 PLUS $1.50 POSTAGE AND HANDLING TO: EDP PROFESSIONALS DEPT-BB 2596 DE FOORS FY NW; ATLANTA, GA 30318

Circle 52 on inquiry card.

CHEAP TALK modems

103 Type Full Duplex 0-300 Baud
RS 232 Interface, Manual or Lease Line
4 1/2 x 5 PCB Construction
One Year Warranty

$115.00



PHONE 1

Send Check or C.O.D.
Shipping, Handling and Sales Tax Included

Phone 1
1330 East State St.
Rockford, Illinois 61108
Phone (815) 962-8927

Circle 115 on inquiry card.

TTY $395.

KSR-33

W/O COVER, USED
SEND CERTIFIED CHECK
$395+$10 SHIPPING TO:
QSS 888 SPINNING RD.
DAYTON, OHIO 45431



The notes supplied by Peter Skye in May 1977 BYTE (page 68) created a flurry of correspondence activity from numerous sources. One of the best proposals we've seen is that provided by Glen A Taylor in his letter titled "Language Development: A Proposal." The main theme of his ideas is proposal of what might be called a personal computers language development society. For our part, to help foster such efforts, we will provide a "Languages Forum" platform for individuals wishing to participate in print with ideas on personal computing languages. This forum is open to all who have technical contributions or suggestions to make in the field of language design for personal computing systems.

A fundamental ground rule is that persons submitting letters should supply a complete address and be willing to correspond with other readers. Telephone numbers will be printed if authors of letters to this forum supply them and indicate a willingness to get together via that medium.

Language Development: A Proposal

Glen A Taylor
The Wisconsin Research and Development
Center for Cognitive Learning
University of Wisconsin
1025 W Johnson St
Madison WI 53706

After reading Peter Skye's note in May 1977 BYTE and exchanging correspondence with him on the subject of a high level language for personal computing, I am moved to offer the following comments and suggestion. I have two fears. My first is that BASIC may become for home computing what FORTRAN is for large computers, an anachronism which is *the* defacto programming language. My quarrel with both these languages derives from the following. They are vast improvements over the tedium of programming in assembly language. They are sufficiently powerful to allow most problems to be solved. They are almost universally available. Herein lies their insidious threat. For all these apparent benefits, the programmer still pays an invisibly high cost in their lack of well-structured syntax. Programs cannot be given good clear logical structure as an automatic consequence of the language; only rudimentary mnemonic naming and labelling are permitted; and large amounts of fairly tedious detail must still be attended to in coding reasonably complex programs. Of course, I'm simply restating the often heard arguments for structured programming, but it is a concept gaining rapid widespread acceptance in mainstream computing.

My second fear is that people who feel as I do, that BASIC is simply not good enough to be enshrined for the next 25 years, will endeavor to supplant it with their favorite programming language. I'm not denying the propriety or utility of efforts to implement APL or PL/I or even good structured programming languages such as PASCAL. There is room for several languages in home computing, but I cannot see any of these "large computer" languages as the best choice for a standard home computing language. None of these languages is without flaws. More importantly, none was written with the needs and limitations of home systems and hobbyist programmers in mind. We must not allow our preferences and prejudices to influence our thinking about what is appropriate and necessary for this new computing field.

My suggestion is that a group be formed for the purpose of defining a suitable personal computing language. I see this as a unique opportunity and high moral responsibility. We are actively engaged in developing a technology that promises to touch the lives of millions of people who are as yet naive to computing. What finer ambition than to develop a language that is human oriented, powerful, flexible, and that is well-suited to the capabilities of home systems for the forseeable future. We are fortunate that there are years of research into programming languages and a vast store of programming concepts at our disposal. We need not fashion a language of dated language concepts and practices. We absorb state of the art hardware technology as soon as it is marketed. We should lead the computing field in readily utilizing state of the art software technology.

Therefore, I challenge readers of BYTE to take the lead and place their support behind such an effort. Here too there are valuable lessons to be learned from the successes and failures of similar ventures

in the mainstream computing field. The development of such a language must not be delayed until there is little chance of displacing a firmly entrenched BASIC. The effort must enlist the support and assistance of several of the major manufacturers who are committed to offering the language as part of their major software line and providing continuing support for it. Finally, the services of a group of people who have experience with present home systems, a clear vision of where the field is most likely to go, and an expert knowledge of modern language design must be enlisted.

I hope you will consider this suggestion. I hope the readers of BYTE will provide vocal support for this idea, thereby encouraging you to support such a project and demonstrating its ultimate economic feasibility to those who would have to support its cost. I am almost certain that you will find the persons with the necessary technical qualifications to serve on the language designing group among your readership. I challenge these persons to step forward. ■

Comments on Peter Skye's Language Proposal

Jeffrey L Kenton
Consultant
One Bacon St
Wellesley MA 02182

Peter Skye's proposal to develop a higher level language for microcomputer use is a fine idea, but it seems to be going astray. If the project goes forward as described in the May 1977 Technical Forum it will be an expanded PL/I with added features from APL and SNOBOL and an apparently huge character set. It appears it was planned to be all things to all people (a replacement for all general purpose languages), and I think it will fail for that reason.

Programming languages have been developed to meet particular needs, and they can best be judged on the power and appropriateness of their constructs for dealing with the intended class of problems. SNOBOL, TRAC and LISP do arithmetic poorly but are quite powerful when dealing with strings and natural languages (English, for example). RPG, despite its somewhat primitive nature, is widely used because it is simple and oriented specifically towards producing business reports. (The business world would be far more interested in RPG running on a micro than anything else I can think of!)

The proposed PL/Skye will make no one happy. The comment that nothing a particular language can do can't be done in PL/I misses the point. BASIC is simple and interactive; APL is powerful and elegant; PL/I is a poor substitute for either. PL/I is fine in large EDP shops which want to convert all their FORTRAN and COBOL programmers to a single, powerful language. It doesn't need APL as a subset. Furthermore, it might pay to remember early experience with PL/I. The first compilers produced atrocious object code, and some of the features never did work. It took compiler writers quite a while before they learned to produce accurate, optimized

code; it will be far worse on a microcomputer. Nor is writing the PL/Skye compiler in PL/Skye a practical idea. Intel has written an 8080 resident PL/M compiler in PL/M which requires well over 100 K bytes of code to run (and a disk operating system which supports overlays). An equivalent assembly language version fits in 12 K.

The first question to ask when beginning a large project is, "What am I trying to do, and what is it going to be good for?". If you haven't answered this question you can never tell when your program is finished, nor whether it works. I'm afraid that's the case with PL/Skye, and we may shortly see a programmer jump on his horse and go riding madly off in all directions. ■

Notes on Floating Point and Critique of PL/Skye

Stephen R Alpert
Assoc Prof of CS, WPI
Vice Chairman, SIGMINI (ACM)
11 Ridgewood Dr
Auburn MA 01501

I would like to add a few comments on the articles that appear in May 77 BYTE.

1. In Sheldon Linker's article "What's in a Floating Point Package?", page 62, there are a few items that should be mentioned.

a. Usually one tries to keep all floating point information normalized. Let's consider an example: let the exponent be base 10 and assume we have four decimal digits of storage. Then

$$0.0025 \times 10^5 = 0.2500 \times 10^3 = 250$$

Clearly I have a choice of storage. But what about 2576? Then I can only use 0.2576×10^4. Chances are if my operands aren't normalized then the result may not be also.

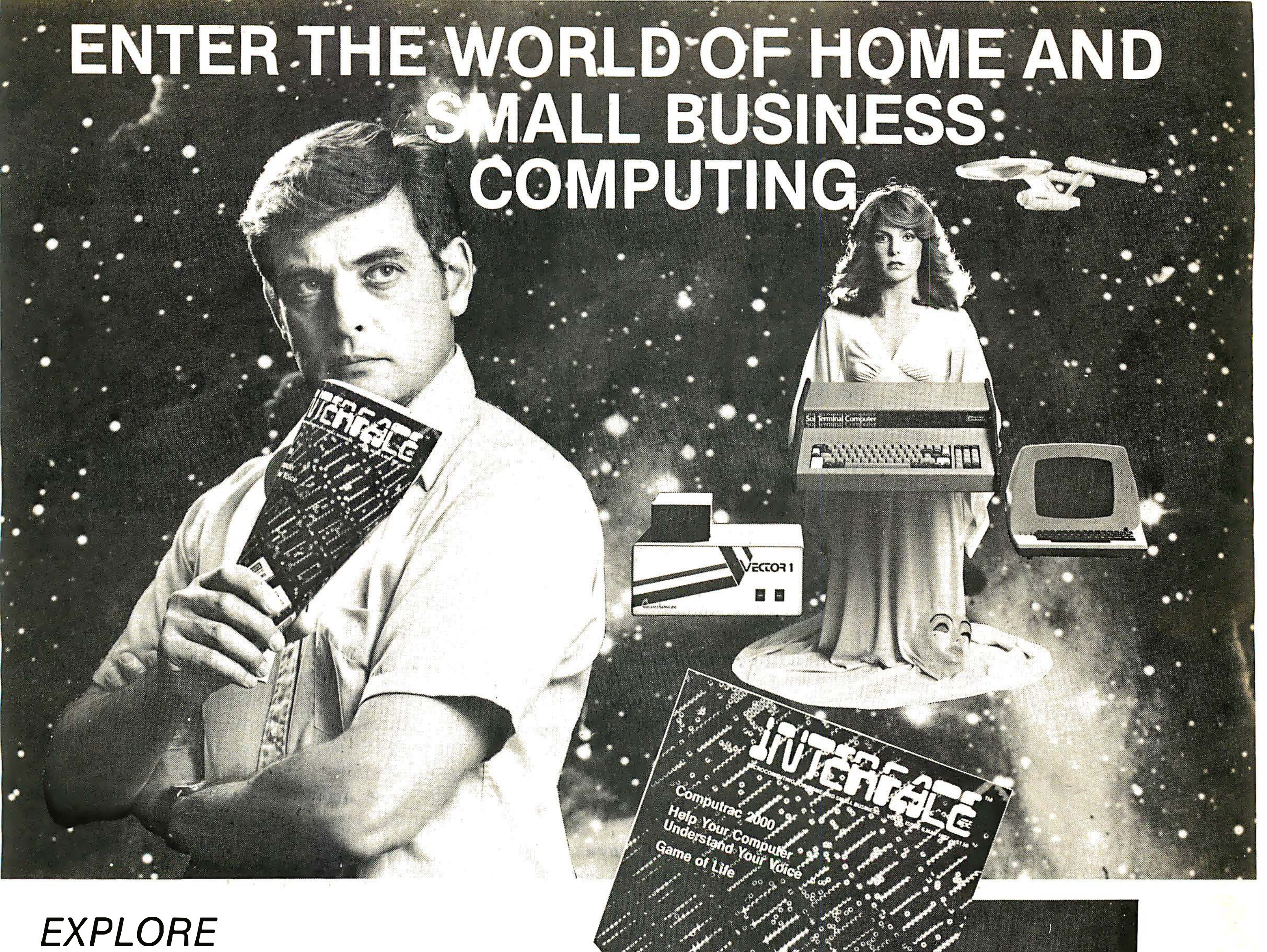
b. In conjunction with normalized data, a hexadecimal base will yield a larger range than a binary base, but it will not carry the significance of a binary base. Hexadecimal base means that a leading hexadecimal digit of 1 will waste three binary bits!

c. In all my years of computing (14) I have never had a need for numbers greater than 10^{38} except for the legendary $24 Manhattan Purchase at 6% for 300+ years. I would suggest the following compromise.

32 bits = 4 bytes		
EXP	S	MANTISSA
8 bit exponent	↑ algebraic sign	23 bit mantissa

The exponent is a two's complement (excess 200) binary exponent. The dynamic range is 10^{76}. The sign bit is stored in place of the normalized most significant bit of the mantissa. Simple shifts or tests will determine the sign (and hence insertion of the MSB is easy).

2. Is Peter Skye serious? I just finished an 8 month project (on the side) writing a compiler for a pseudo-subset of PASCAL. It was a real job. He will require the user to have 32 K bytes just to load the compiler.



EXPLORE

INTERFACE AGE

MAGAZINE

★ MONTH AFTER MONTH LOOK TO **INTERFACE AGE MAGAZINE** FOR THE LATEST INFORMATION ON THE DYNAMIC WORLD OF PERSONAL COMPUTING.

- Use your personal computer for auto repair, work bench controller, teaching machine, central information bank and design test center.
- Control your small business with your own real-time accounting and inventory control system.
- Set your computer to turn sprinklers on and off, manage a household security system, feed your dog.
- Establish a recipe bank to plan daily meals and generate its own shopping list.
- Evaluate the stock market, set up gambling and probability programs. Evaluate odds on sporting events and horse racing.

★ ARTICLES RANGE FROM THE FUNDAMENTALS OF COMPUTERS TO LANGUAGES AND SYSTEM DESIGN. APPLICATIONS INCLUDE BOTH PROFESSIONAL AND NON-TECHNICAL.

★ READ **INTERFACE AGE** FOR THE LATEST ON NEW PRODUCT INFORMATION AND TECHNICAL BREAKTHROUGHS.

- May's issue included inside the FLOPPY ROM™ — a vinyl record which is played on a conventional phonograph to enter this month's program in your computer.

★ ORDER YOUR SUBSCRIPTION NOW!

12 Monthly Issues: $14 U.S., $16 Canada/Mexico, $24 International

Name ____________________

Address ____________________

City ____________________ State ________ Zip ________

☐ Check or M.O. (U.S. Funds drawn on U.S. Bank) ☐ Visa Card ☐ Master Charge

Acct No. ____________________ Exp. Date ________

Signature ____________________

Make checks payable to: **INTERFACE AGE MAGAZINE**, P.O. Box 1234, Cerritos, CA 90701

Circle 74 on inquiry card.

3. A suggestion: following the example of the *Communications of the ACM*, unless programs are for specific hardware tasks, they should be written in a single "standard" language. My current choice would be PASCAL for the following reasons:

a. It has a very strong (precise) standard. Anyone can purchase a user's manual and report from Springer-Verlay for about $6.

b. There is a strong user's group that is international in scope.

c. The language permits definition of user defined data types. One could add bytes, bits, etc. Pointers are standard constructs in PASCAL.

d. If a standard were expounded, I'm sure that in short order actual compilers would soon appear.

e. A top down (or recursive descent) compiler for PASCAL is made easier if the output is, in fact, an assembly language source. This output can then be fed to your favorite assembler. Additionally, by using PASCAL type switches one could imbed assembler code directly into the higher level code.

f. PASCAL programs would then be highly portable, enhancing the standard even more.

Lest you think PASCAL is my only language, I have also used and taught most of FORTRAN, ALGOL, APL, LISP, FOCAL, BASIC, BLISS and SNOBOL (and a little PL/I).

I hope this letter stirs the pot a bit.

The only problem with making a highly desirable standard representation for published programs is the problem of actually achieving the representations in that form. Documentation of an adequate "representation language" is a necessary first step to a highly desirable end. A syntax and semantic checking program (a compiler minus code generation) would also be most useful from a publication's point of view to verify and correct superficial details of programs. But such a standardization also requires authors and designers literate in the language as well. Would anyone care to make further comments on this subject of adopting a representation standard for programs in print?■

What's Wrong with PASCAL, Mr Skye?

David A Mundie
104B Oakhurst Cir
Charlottesville VA 22903

I am writing in response to the ongoing dialogue in your pages over the choice of a high level language for microcomputers.

Mr Crone's analogy with English (May 1977 BYTE, page 112) is misguided. English, though archaic, is both beautiful and well-suited to its purpose; FORTRAN is neither. His letter conjures up visions of our grandchildren using dream computers, yet still struggling with format statements and amorphous programs simply because we lacked the courage to junk our outdated languages as readily as we junk our outdated machines. They will curse us for it.

I do not design computers, so perhaps I am missing something, but Mr Skye's comment on PASCAL (May 1977 BYTE, page 68) puzzled me. The point is not that PASCAL does nothing other languages can't do; the point is rather that PASCAL does virtually everything the other languages do, but starts from a much simpler set of basic constructs. I should have thought that sort of efficiency was just what was needed for microprocessors.■

Questioning APL

Rich Snodgrass
229 Llano Dr
Portland TX 78374

I greatly enjoyed the August 1977 issue of BYTE on APL. The articles were well-done and contained much useful information.

I do wish, however, to take issue with some of the views expressed by E H Anthony in the Technical Forum. I became weary with superlatives such as "one of the greatest intellectual achievements of this century," "the teacher of the century," and "computer languages scarcely bear close comparison with APL." I hear similar comparisons every year when Detroit comes out with a new model.

Such statements are subjective by nature and hence a total matter of opinion. However Mr Anthony's statement that APL is the most "general-purpose mental tool"

in comparison with other computer languages is just too sweeping to let pass without comment.

Generality is an important criterion in judging programming languages and, to a limited extent, APL is blessed in this regard. However, when all the features of APL are examined, it is rather specific.

For example, only homogeneous multidimensional arrays, programming of numbers, and single characters are allowed as data structures; COBOL's heterogeneous arrays and the list structures available in LISP and SNOBOL are completely lacking. Formatted IO and external data files are not specified in the language definition, features found even in lowly FORTRAN. Structured programming is very difficult in APL, and even the most basic control structures are missing [except, of course, for the computed (GOTO)].

Ironically, that "regrettable language" mentioned in the article, PL/I, has *all* the features listed above. PL/I also excels in readability and run time efficiency, especially in comparison with APL. Now PL/I is not even close to the "perfect computer language," although it does have more generality than many other languages, including APL.

It will now hopefully be evident that no computer language is best at everything, even APL. The incredible variety of tasks that the computer is now given makes it impossible for one language to be proficient at them all. System programmers should keep an open mind when deciding which languages to implement: LISP, SNOBOL and ALGOL, as well as several other important languages, can be implemented on microcomputers with reasonable memory requirements. All it takes is someone to do it.■

Suggestions for APL Optimization

Jon D Roland
Micro Mart
1015 Navarro
San Antonio TX 78205

There seem to me to be three important difficulties with APL that are unnecessary, and that might be corrected in the development of APL for microcomputer systems.

The attractiveness of APL arises not only from the efficient coding and powerful primitives it provides, but from the ways it facilitates *interactive* programming, so that the computer user can write programs

on the terminal with a minimum of preparation on paper.

Interaction

The first difficulty arises from the fact that APL is entered from left to right, but executes from right to left. I, like many users, do not always know when I begin a line of program how I am going to finish it, which means that if I am at the terminal, I must either make frequent corrections to what I have already entered, or prepare the statement on paper before I key it into the terminal. I see no fundamental reason why APL could not be reversed, or a reverse APL made an option for the convenience of programmers who think in RPN. There would be no need to change the character set; just make execution from left to right.

Character Displays

The APL character set is not ideal for use with 5 by 7 dot matrix printers or video displays. Some overstruck combinations are not readily distinguishable. Could we not choose characters that are optimal for legibility and aesthetic appeal, even when overstruck?

Keyboard Layout

The arrangement of APL characters on the keyboard is not convenient for rapid, error-avoiding typing. Why could not the APL characters be arranged in some pattern that is optimal for the user who wants to touch-type his input, as the Dvorak keyboard is for ASCII characters (except the special command keys).

If APL, in some form, is destined to become a kind of universal high level computer language, then let us avoid features that are unnecessarily cumbersome for interactive usage, and resolve now to develop a language that is optimal in practical terminal interaction. Let's not make a mistake like the QWERTY keyboard!■

Some Comments on "An APL Bigot Speaks"

Henry Brandt
Ithaca NY 14850

In reference to the letter from Gary Luther in the August 1977 BYTE, page 12 ("An APL Bigot Speaks"), I would like to offer a few points of clarification.

First, the European APL implementation that he speaks of is described in the *IBM*

Systems Journal, volume 16, number 1, and is entitled "An APL Interpreter and System for a Small Computer." The authors of this paper took a full APLSV interpreter and broke it up into 289 128 word modules which are paged into main memory of a System/7. This technique should still prove popular among hobbyists for whom processor costs are overshadowed by the cost of large amounts of main memory.

Second, the IBM 5100 doesn't really put the full APL language in 16 K, as Mr Luther indicated. The 16 K to which he refers is the user workspace, which is available in 16 K increments up to a maximum of 64 K. The APL interpreter is resident in 108 K bytes of read only storage. I suspect a commercially available ROM offering of this nature is still a number of years away.

Lastly, unless we see dramatic changes in the cost of memory, we are most likely doomed to implementing a subset of APL in either ROM or a complete version of it in an overlay fashion for those who possess secondary storage devices such as floppy disks.■

Programming Quickies

A 6800 Program Relocator

Andrew A Carpenter
POB 841
Gordonsville VA 22942

Here is a short program relocator that may be of interest and use to readers of BYTE. The program to be relocated must presently reside in memory. Hexadecimal addresses A002 and A003 are set to the address of the program. Addresses A004 thru A007 are set to the beginning and ending addresses of the new location for the program. This program was written for a SwTPC 6800 system.

```
LOCN B1 B2 B3 >
1000          >
1000 FE A0 02 >        LDX  A002
1003 A6 00    >        LDA A00, X
1005 08       >        INX
1006 FF A0 02 >        STX  A002
1009 FE A0 04 >        LDX  A004
100C A7 00    >        STA A00, X
100E BC A0 06 >        CPX  A006
1011 27 00    >        BEQ  1019
1013 08       >        INX
1014 FF A0 04 >        STX  A004
1017 20 E7    >        BRA  1000
1019 3F                SWI
1012 06       >
101A
***  UNRESOLVED ITEMS:

***  SYMBOLS, SORT?■
```



Relocatable Object Code Formats

In the July 1977 issue we published a document handed out into the public domain by Peter Formaniak and David Leitch of Mostek. (See "A Proposed Microprocessor Software Standard," page 34, July 1977 BYTE.) Our purpose for publishing the document was to get some interchange started on the issue of relocatable object code formats.

In this continuation of the discussion of the subject of relocatable formats, we have three items. One is a letter reacting to the published information and making some suggestions. The second item is a format used by Technical Design Labs, originated by Neil Colvin. This text was given to us at the TDL booth at the National Computer Conference in Dallas last June, and offered as documentation of a standard which is in use by that firm, and is reportedly being examined for adoption by two other major software vendors in the personal computing marketplace. The third item is a letter from Tom Pittman critiquing the TDL standard, an item which resulted from a recent phone conversation.

As an addition to the discussion, the note following Tom's critique was received from Philip Tubb, and has a bearing on the process of compiling and making available standards documentation for this field.

A Response to "A Proposed Microprocessor Software Standard"

Carol Anne Ogdin
100 Pommander Walk
Alexandria VA 22314
(703) 549-0646

The proposal put forth by Formaniak and Leitch is certainly a step in the right direction, but it also sets unreasonable limits on the lengths of symbols permitted. By imposing a limit of six bytes on symbol length, the authors propose to throw back programming techniques to the 1960s. A simple analysis of their standard shows a clear and obvious format that permits symbols of virtually unlimited length, although an imposition of a length limit of (say) 64 bytes would not be unreasonable.

In record types 02 and 03, I propose the following modification of their conventions:

Byte Number	Description
1	Dollar sign ($) delimiter
2, 3	Length of the symbol (or zero,

implying the symbol is terminated by carriage return or other control code)

4, 5	Most significant byte of the address (definition for record type 02, address of chain for type 03)
6, 7	Least significant byte of the address
8, 9	Record type (02 or 03)
10. . .	Symbol text
Last 2 bytes	Checksum
CRLF	Carriage return, line feed (Delimiter of end of record and end of symbol text)

The advantage of this format is that it permits (but does not require) longer symbols. If the particular assembler author needs to impose some arbitrary restriction on mnemonic and symbolic names, so be it. But, to impose such arbitrary restrictions in a proposed standard assures that the standard will not be adhered to in practice.

Finally, a note about proposed standards themselves. Unless and until the personal computing movement gains a coherent voice through a single forum, standards will remain nonstandard. It behooves the users to get behind the standards movement. Unfortunately, the ANSI mechanism is too burdensome for our needs. If some enterprising publisher (hint, hint) were to dedicate a half a page to listing the currently accepted user standards and the references where the final definition can be found, it might begin to serve as that needed central forum. Could such a list be published every couple of months or so? I should point out that without such a single point of reference, proposed amendments (and general acceptance of the original or amended proposal) will never get properly promulgated to the necessary readers.■

Technical Design Labs Relocatable Object Module Format

Neil Colvin
Technical Design Labs
Research Park Bldg H
1101 State Rd
Princeton NJ 08540

DEFINITIONS

Object Module: The output from a language processor. Object modules may be loaded into memory for execution at fixed addresses.

Relocatable Object Module: An object module containing information which allows the loader to place it anywhere in memory address space.

Internal Symbol: A symbol whose location is available to other modules besides the one in which it is defined.

External Symbol: A symbol which is used in a module but is defined as an internal symbol in some other module.

Entry Point: An internal symbol in a module which is used to select the module for loading as a

result of its being referenced in another module as an external symbol.

Linkable Object Module: An object module containing information identifying external, internal, and entry point symbols which can be "linked" to other similar modules by the loader.

Relocation Base: The external symbol whose address is the base for the relocation of an object module. The external symbol may represent a program, data, or common area of memory.

Object Module Format Definition

The object module format is an extension of the Intel "hex file" format, but is not compatible with that format. The module consists of a sequential file of ASCII characters representing the binary data, symbol and control information required to construct a final program from the module. All binary bytes within this structure are represented as two ASCII characters corresponding to the hexadecimal value of the byte (eg: 11001001 → C9). All ASCII values are represented by the corresponding ASCII character (eg: A → A).

Each of the different records within the module is indicated by the use of a prompt character as the first character of the record (in the Intel format, this is the ":"). The valid prompt characters are:

Character	Meaning
!	module identification record
@	entry point record
#	internal symbol record
\	external symbol and relocation base record
&	symbol table record
;	data or program or end of file record

Every record in the module is terminated by a one byte binary checksum of all of the preceding bytes in the record except for the prompt character. The checksum is the two's complement of the sum of the preceding bytes. Either output format (two character binary or one character ASCII) still counts as only one byte in the checksum (ie: before conversion for output).

In addition, each record is preceded by a carriage return and line feed sequence to facilitate listing the module on an external device.

- **Module Identification Record ("!")**

Byte Number	Description
1-2	CR/LF
3	Exclamation point (!) prompt.
4-9	ASCII module name. *[See comments on length in letter by C A Ogdin.]*
10-11	Checksum.

- **Entry Point Record ("@")**

Byte Number	Description
1-2	CR/LF
3	At sign (@) prompt.
4-5	Number of entry points in this record.
6-??	ASCII names of entry points, six bytes per name. The names are left justified and blank filled.
??	Checksum.

- **Internal Symbol Record ("#")**

Byte Number	Description
1-2	CR/LF
3	Pound sign (#) prompt.
4-5	Number of internal symbols in this record.
6-11	ASCII name of internal symbol, left justified and blank filled.
12-13	Relocation base for symbol. The value of this symbol is relative to the relocation base specified.
14-17	Symbol value (16 bit).
****	The above three fields are repeated for each internal symbol in the record.
??	Checksum.

- **External Symbol and Relocation Base Record ("\")**

Byte Number	Description
1-2	CR/LF
3	Backslash (\) prompt.
4-5	Number of external or relocation symbols in this record.
6-11	ASCII name of the symbol, left justified and blank filled.
12-13	Relocation number assigned to this symbol in this module. This number is unique for each symbol. It starts with one and increases sequentially for each subsequent external or relocation base symbol.
14-17	Relocation segment size or external reference flag. If this value is zero, it represents a reference to a symbol defined externally to this module (usually a subroutine or global data item). If it is nonzero, then the value is the size of the relocation segment as

	defined in this object module. This segment can contain either code or data, and may be located anywhere in memory by the loader, independent of any other segment.
****	The above three fields are repeated for each symbol contained in this record.
??	Checksum.

- **Symbol Table Record ("&")**

Byte Number	Description
1-2	CR/LF
3	Ampersand (&) prompt.
4-??	The remainder of this record is identical to the internal symbol record. All symbols defined in this module are contained in these records.

- **Data/Program Record (";")**

Byte Number	Description
1-2	CR/LF
3	Semicolon (;) prompt
4-5	Number of binary data bytes in this record. The maximum is 32 binary bytes (64 bytes of ASCII representation). If this value is zero, this record is a end of file record, described below.
6-9	Load address of the data relative to the specified relocation base.
10-11	Relocation base for all relocation in this record. All relocatable values in this record are added to the current value of the specified relocation base before being put into memory.
12-13	Relocation control byte. This byte controls the relocation of the next eight bytes in the record (if that many remain according to the count field). The bits are used from left to right. The bits have the following meanings:

0: a single absolute byte implies load unmodified.

10: a two byte relocatable value, least significant byte first implies add the 16 bit value to the current relocation base, and load the result least significant byte first.

110: a three byte reference to a different relocation base. The first byte is the relocation base number, and the two after that are the 16 bit value, least significant byte first. This implies add the specified relocation base to the 16 bit value, and load the result least significant byte first.

	Note that a two or three byte combination is never broken across a record boundary.
14-29	Data bytes controlled as above.
30-??	The above control and data byte combinations are repeated as specified by the count.
??	Checksum.

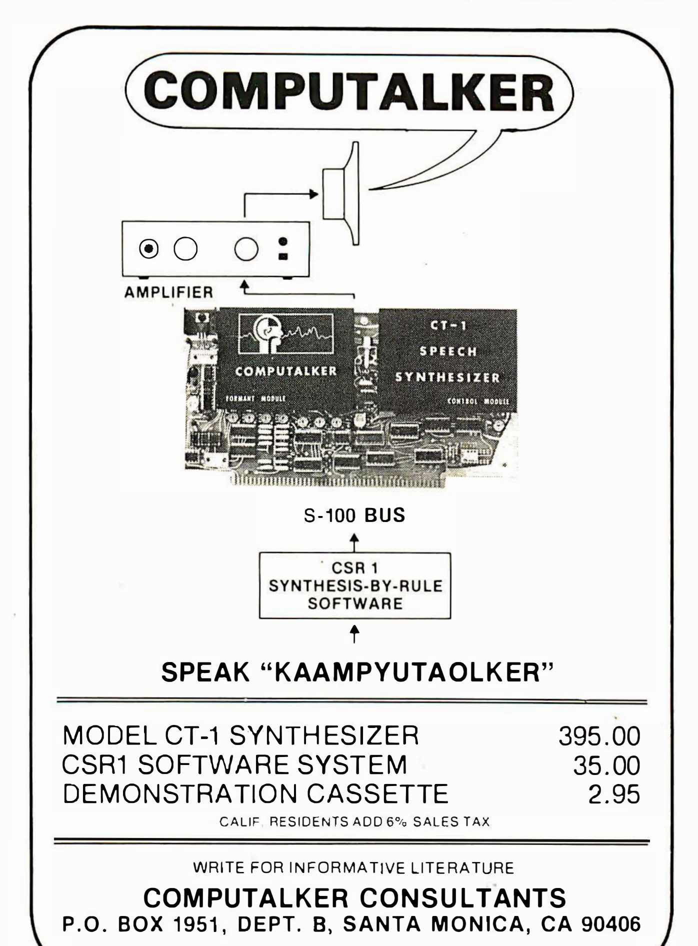
- **End of File Record (";")**

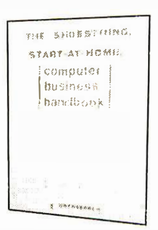
Byte Number	Description
1-2	CR/LF
3	Semicolon (;) prompt.
4-5	Zero to indicate end of file record.
6-9	Starting address for module relative to the specified relocation base. This address is optionally generated by the language processor, and may be zero.
10-11	Relocation base for starting address.
12-13	Checksum.

Relocation Bases

One of the important capabilites of this object module format is the ability to specify multiple relocation bases for the module contents. These relocation bases may represent ROM versus user programmable memory shared common areas, special memory areas such as video refresh areas, etc. Within a module, each of these relocation bases is assigned a name, and implicitly, a sequentially generated number. The relocation bases are actually assigned values at load time, but all memory references within the module are made relative to one of these bases.

Four of the relocation bases (0 to 3) have predefined names and meanings, and are treated differently at load time than the remainder of the bases. Base 0 represents absolute memory locations (ie: It always has the value 0). Base 1 has the name ".PROG." and represents the program area (may be ROM or PROM). Most program code is generated relative to this relocation base. Base 2 has the name ".DATA." and represents the local data areas for each module. Most local data is defined relative to this base. Base 3 has the name ".BLNK." and represents the global "blank common." This relocation base is always assigned the value of the first free address in memory after the local data storage (.DATA.) and other data relocation segments. Because it is





always the last allocated, modules referencing this area can be loaded in any order, regardless of the amount of the area they use.

Relocation segments relative to bases 1 and 2 (.PROG. and .DATA.) are always loaded additively. (ie: After each module is loaded, the value of the relocation base is increased by the size of the segment.) All other relocation bases are assumed to have constant values during the load process and may be allocated by the loader.■

Comments on the TDL Relocatable Loader Format

Tom Pittman
Itty Bitty Computers
POB 23189
San Jose CA 95153

It begins to look like we are going to see the same diversity in design of software in the personal computer industry that we have seen in the hardware design. This remark is prompted by a document describing Technical Design Labs' "Relocatable Object Module Format" which I recently had a chance to examine.

TDL is not the first to promote a relocatable format, and you may be sure they will not be the last. Let me suggest some reasons. But first I should remark that the people at TDL have obviously put a lot of thinking and work into their format. It will serve them for much software, some of which is clearly still in the future. My personal impression is that the format tries so hard to be "efficient" that it has acquired the distinct flavor of a kluge, but I will admit that to be a matter of taste and not a matter of substance.

The problem with the TDL format, and also with the other formats which have come before, is that it is limited to the relocation of 16 bit addresses. This may be satisfactory for relocating jumps and subroutine calls, but it is quite unworkable for data references where the actual address of the reference must be computed from a relocated base address plus some computed offset. It is true that you can use an LXI instruction in the 8080 or Z80 and do the arithmetic through the register accumulator ADD instructions, but in the 6800 there is no convenient way to do arithmetic from an address loaded into the index register with an immediate mode. Even worse, the 6502 has no 16 bit register which may be loaded immediate, and the programmer would be forced to such subterfuges as defining an address constant containing the relocated address, then using extended addressing to refer to it. Another hazard which does not affect the 8080 and Z80 is the problem of relocating base page addresses. So far I have seen nobody address this problem, and yet the 6502 is effectively inoperable without reference to page 00. Are we to continue to force users to laboriously allocate page 00 even after relieving them of the same drudgery with respect to the rest of memory?

I should also like to mention two other problems which have not been addressed, but which are considerably less severe. One has to do with the problem of the difference between two relocated addresses. Most assemblers do not allow constructs of the form (LXI B, ALPHA-BETA), where ALPHA and BETA are both externals. The

construct is not allowed because there is no way to pass expressions to the loader. It is a useful construct, and at present the only way to accomplish the same effect with a relocatable code costs seven extra instructions. But as I said, this is less important. More important is the problem of error checking. For reliable media, who cares? But if we are going to bother to put checksums in the format, we should be sure that everything important is checked. As far as I know, only the hex absolute format defined by MOS Technology does this, unless the TDL loader insists on the presence of the carriage return linefeed and requires the next character to be either a colon or dollar. Most loaders simply ignore all text until the header character is recognized, which gives rise to the possibility that lines may be dropped, an occurrence I know to have happened. I think the loader should ignore control characters (CRLF should be optional) but have some safety against dropped lines.

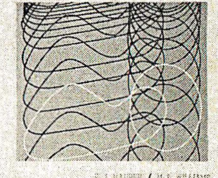
I said we would be seeing several relocatable formats. Like the hardware designers, no software designer is completely satisfied with what someone else has designed, so he/she wants to do her/his own. But more than that, when a proposed standard has serious deficiencies, it will not be widely accepted. As you no doubt have suggested by now, I think I can do better. Time alone can tell whether we actually achieve any standards in this area.■

Announcing the Central Standards Library

To help solve some of the standards problems in the small computer and microcomputer field, ALF Products is sponsoring a Central Standards Library (CSL). After discussions with several manufacturers in this field at the West Coast Computer Faire, ALF has set up the CSL as a means of standards information exchange for manufacturers, consumers, hobbyists, and others interested in standards. The Library will collect submitted standards and distribute them on a nonprofit basis. For more information on available standards, on how to submit standards, and on the Library's services, send $1 (to cover printing and mailing costs) to The Central Standards Library, c/o ALF Products Inc, 128 S Taft, Denver CO 80228. You will receive a copy of the first CSL newsletter and the first submitted standard (a parallel interface standard). Manufacturers currently participating include: ALF Products, IMSAI Manufacturing, PolyMorphic Systems, Proko Electronics, Vector Graphic, and Video Terminal Technology.■

Book Reviews

A Collection of Programming Problems and Techniques *by H A Maurer and M R Williams, Prentice-Hall Inc, Englewood Cliffs NJ, 1972, 256 pages, 6 by 9 paperback. $6.95.*



Among the things I like about the computer field is that it inspires a new style of solitaire: It's me against the computer, and, if I'm persistent, I can always win. A frequent problem, however, is finding a game which will both bring satisfaction and sharpen my skills. Many beginning programming books give only modest examples and problems which do not challenge the intermediate student. Since the trip from apprentice to journeyman is paid for only with experience, a good selection of programming problems is a must.

For the enthusiast seeking a challenge, or the novice wishing to become a pro, Messrs Maurer and Williams have filled this need with nearly 400 problems of varying degrees of difficulty. These exercises provide experience in most of the common problems encountered by programmers. Working your way through the book will provide an insight into the mysteries of applied higher mathematics, even though no knowledge of mathematics above the high school level is required. You'll find sections on number theory, random numbers and equations in one variable. The chapter on games discusses chess and checkers, and there are number games throughout the book. The IO section challenges you to print bridge hands, or perhaps a calendar. You can make maps, circles and family trees. Some of the great problems and legends of history are also described; perhaps you can solve them.

The problems are couched in general terms so that any of the common programming languages may be used. Introductory problems range from the reading and printing of data to the calculation of a bowling score. More difficult problems address satellite orbits and language translation. There is an excellent advanced section dealing with simple compilers and three-dimensional plotting, as well as the sorting and merging of data. In working out these problems the programmer will gain a facility in common applications.

An appendix of partial answers to prob-

lems is provided. Since there are many ways to program a given task, sample programs are not given. A well thought out index provides a reference to any particular problem or concept.

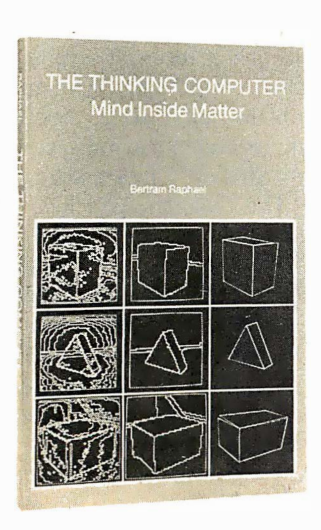
The excellence of this book lies not only in its graduated problems, but also in the truly great variety of the exercises. It is not an introductory text, but of course there are many of those. It does promise to make the reader a "tackle anything" programmer, which is the very best kind.

Noel K Julkowski
18755 Van Buren St
Salinas CA 93901■

The Thinking Computer: Mind Inside Matter *by Bertram Raphael, W H Freeman and Company, San Francisco, 1976. Softbound $6.95.*

This excellent book is perfectly suited for the technically inclined reader who wants to know more about artificial intelligence (AI) and robotics. Written by one of the pioneers in AI research, it provides comprehensive, up-to-date coverage of the field in a style that adroitly balances technical depth against readability and understandability. Raphael is especially effective in illustrating abstract ideas with memorable examples, like a "cryptarithmetic" puzzle which is explored via a tree search, and a butler and maid mystery which is solved by theorem proving techniques.

The book's introduction, which provides a basic orientation to computers for the uninitiated, also discusses special AI peripherals and software, and deals, albeit briefly, with two common misconceptions about computers: that they are just giant arithmetic calculators, and that they are dumb



mechanical servants that can do only what they have been explicitly told how to do. Raphael's refutation doesn't anticipate possible further objections, but these issues have been discussed elsewhere.

The first technical topic is the representation of information about problems, which has proven crucial in AI application software. Strings and list structures are described and applied to the representation of board games, symbolic algebraic formulas, English sentences and pictures.

The next topic is search. Breadthfirst, depthfirst and progressive deepening strategies are applied to the problem of searching trees, and techniques for adding knowledge to the search are described. The problem of finding the shortest route between Paris and Vienna is used to illustrate the search for a path through a general graph structure. Techniques for searching game trees, including evaluation functions, minimaxing, and alpha-beta pruning, are briefly described.

A major chapter deals with pattern recognition and theorem proving techniques. The latter discussion presents Wang's algorithm for the propositional calculus, the undecidability of the predicate calculus and its implications, resolution theorem proving and answer extraction (with a beautiful example, "Dr Coleman's wife"), and other formal systems such as modal, probabilistic, multivalued and fuzzy logics. While it is sometimes cursory and far from rigorous, this is easily the most readable approach to a sometimes forbidding topic that I have ever seen; every reader will appreciate it.

The presentation of formal problem solving methods is followed up by a discussion of informal approaches. The paradigm used in Newell and Simon's General Problem Solver is presented and applied to the "frame problem," that of updating a description of the current situation as actions are taken, with an illustration of a robot moving through a room. The possibility of applying theorem proving techniques to the frame problem leads to a description of STRIPS, the problem solving system used in the robot "Shakey" at Stanford Research Institute. The chapter concludes with a discussion of the question, "Can a computer learn?", with illustrations taken from Samuel's checkers playing program and Winston's concept learning program at MIT.

Raphael then turns to computer understanding of natural language, and again provides a technical discussion of the methods without sacrificing reader understanding. Phrase structure grammars, transformational grammars, and the difficulties

with these approaches are described, leading to a consideration of the interplay between syntax and semantics, and more recent approaches such as case grammars and conceptual dependency theory. Actual systems which understand English are reviewed, with a special emphasis on Winograd's very successful program, SHRDLU. The chapter concludes with comments on the promise of current research into speech understanding systems.

Succeeding chapters deal with perception and picture processing, and robot systems. Techniques such as smoothing and sharpening, finding edges and lines, and dealing with light and shadow are described in enough detail to give the reader an idea of how these things are done, with illustrations from the work of Guzman, Huffman and Waltz. The history of robots is reviewed, with examples from Ross Ashby's Homeostat, Grey Walter's tortoises, the Johns Hopkins "beast," the MIT robot arm, and Meredith Thring's inventions. Then a case study is presented of Shakey, the SRI robot.

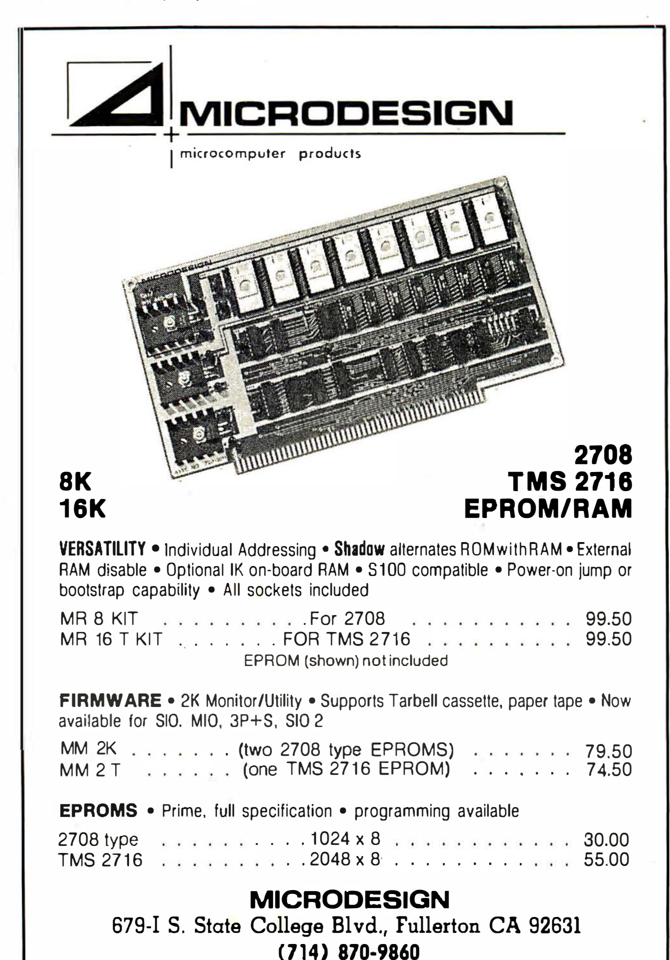
The final chapter comments on "frontier applications" in which the fruits of AI research can be used to better our world. Examples are drawn from work in education, psychology and medicine, as well as other fields. The book concludes with an eloquent commentary on the potential for dehumanization and the promise of enrichment of our society posed by intelligent machines. The key, of course, is understanding, and Raphael has made a real contribution to popular understanding of artificial intelligence research by writing this book.

Dan Fylstra
Hamilton Hall C-23
Harvard Business School
Boston MA 02163■

The Anatomy of a Compiler *by John A N Lee, D Van Nostrand Company, New York, 1974, 470 pages. Paperbound, $11.95.*

John A N Lee, a professor of computer science at Virginia Polytechnic Institute and State University, has written a book that bridges the information gap between the elementary explanations of compilers which are usually found in the last chapters of introductory textbooks, and the very abstract theoretical explanations, ie: those that speak in terms of "given a set S." Dr Lee's clear, precise prose possesses a great deal of flair leading the motivated reader from first principles to complex operations.

This book is a "how to" book, abundant





in explanation, striving to impress upon the reader the hows and whys of current day symbolic language definition and execution. Dr Lee places a great deal of emphasis upon the differences between compilers and interpreters by emphasizing that compilers produce separate code (object code) that is executed after the compilation phase of execution is finished; but interpreters execute the source code on a line by line basis which may not be optimal in terms of processor time. Compilers, as contrasted with interpreters, output object code to some intermediate storage medium for later execution. This means that execution of compiled programs is often more efficient, in terms of processor time, than interpretive execution each time the program is run. However, source code errors are more difficult to correct in compilers than similar errors in interpreters because there may not be a clear relationship between compiled object code and the original source code. Interpreters, on the other hand, by virtue of their line by line execution characteristic, retain a definite relationship between object code and source code. This simplifies the debugging of source code. As a result of these considerations, we may find an increasing interest in compilers among computer hobbyists as high speed mass storage devices become less expensive.

Dr Lee also discusses, in great detail, lexical analysis and syntactical analysis. He explains that lexical analysis serves to remove redundancy, condense statements and delimit phrases from the source code. Syntactical analysis serves to recognize phrases, parse statements and generate parsed text. After discussing symbol tables which are used by the compiler to reference symbols from the source code, he covers string manipulation and Polish string conversions in great depth. Program control also receives thorough treatment.

Throughout the book Dr Lee draws profusely upon examples of actual implementations of the techniques he describes. Examples are taken from ALGOL, APL, BASIC, FORTRAN, PL/I, and other languages, thereby avoiding the trap of producing a one language book. Also, much to the author's credit, the book is profusely illustrated with flowcharts illustrating the algorithms described. In summary, Dr Lee's book is clear, readable and certainly useful to the serious home computerist. Its wealth of practical information should be welcome to any computerist's bookshelf.

Michael E Sullivan
OZ Division USS Saratoga (CV-60)
FPO NY 09501■

Clubs and Newsletters

HP-65 Users' Club

This club was started to support the HP-65 programmable calculator, but now all modes are supported (HP-25, HP-25C, HP-55, HP-67, HP-97 and HP-65). The newsletter, called *65 Notes,* is an excellent publication in which members (and even nonmembers) share programs, ideas, frustrations, etc. Even if you are not a programmable calculator devotee (timesharers take note. . .) you'll find something here. For more information contact Richard J Nelson, editor, HP-65 Users' Club, 2541 W Camden Pl, Santa Ana CA 92704.

Stock Market Anyone?

An association of persons who have a serious personal interest in using a microcomputer for stock and commodity market investment purposes is being formed. If covariance is more than just another word to you, send a brief note listing your desires, qualifications and market experience/involvement to J Williams, 2415 Ansdel Ct, Reston VA 22091.

PACC

Those of you in the Pittsburgh area should definitely consider getting involved with the Pittsburgh Area Computer Club. There are users' groups, displays, activities and socializing at the monthly meetings. Plan to be there to brag about your system or look at others' systems. Membership is $12 per year. Contact Ed Dehart, president, Pittsburgh Area Computer Club, 400 Smithfield St, Pittsburgh PA 15222, or call Kenn Marks at (412) 352-3412.

Alamo Computer Enthusiasts

This nearly brand new club in San Antonio TX meets on the fourth Friday of every month in room 104, Chapman Graduate Center, Trinity University. Call (512) 532-2340 or 342-3874 for more information. Also, the ACE newsletter needs your input. If you are a brave soul with something interesting on your mind, send your idea to John Stanton, 7517 Jonquill, San Antonio TX 78233, or phone (512) 657-3069.

SPC-12 Users' Group

Anyone who would like to form an SPC-12 users' group in the Chicago area should contact Manuel C Martinez, 7706 W Gregory St, Chicago IL 60656, or call (312) 631-6623.

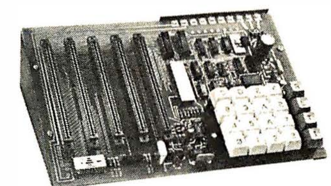
LICA

The Long Island Computer Association is a group of hackers, amateurs and even some pros in the Commack, Long Island area. The monthly meetings feature good speakers, fun and refreshments. The group publishes a newsletter called *The Stack.* Nonmembers are welcome to all meetings; bring the whole family! Write to Long Island Computer Association, c/o Dave Metal, editor, 28 Splitrail Pl, Commack NY 11725.

Washington Amateur Computer Society (WACS)

Every two months or so WACS sends us a rather impressive computer newspaper with items of interest to club members and non-

Conducted by
David Wozmak



members alike. It is printed in the form of a computer printout by a DECsystem-10. The club meets monthly; for specifics, write to Washington Amateur Computer Society, 4201 Massachusetts Av, Washington DC 20016.

6800 Users in San Jose

Anything and everything to do with 6800 microcomputers is of interest. Hardware and software are always on display by area hobbyists, and everyone is welcome. Meetings are on the first or second Tuesdays of each month. Contact the 6800 Computer Club at POB 18081, San Jose CA 95118 for more information.

The Chicago Area Computer Hobbyist's Exchange

The montly publication of the Chicago Area Computer Hobbyist's Exchange is *The Cache Register*. This impressive newsletter has all the necessary club information, some great programs, convention news, editorials and so on. If you want club information, or would like to receive this newsletter, simply write Chicago Area Computer Hobbyist's Exchange, POB 36, Vernon Hills IL 60061, or call (312) 849-1132.

KIM-1

The KIM-1 users' group has introduced *The First Book of KIM*, designed to help beginning KIM users. Introductions to programming, interfacing to KIM, games and utility programs are all covered.

If you are a KIM-1 user/owner/soon to be owner, then the *KIM-1/650X User Notes* is for you. All kinds of helpful software and stuff is packed into this bimonthly. For more information, write *KIM-1/650X User Notes*, 109 Centre Av, W Norriton PA 19401.

South Florida Computer Group

In the Ft Lauderdale/Miami area, meeting times vary. These people put out a newsletter with page numbers in binary. For information on joining the club or receiving the newsletter write to South Florida Computer Group, 1155 NW 14th St, POB 236188, Miami FL 33123, or phone (305) 324-5572, ext 45.

RAMS

The Rochester Area Microcomputer Society is a group of individuals with the

aim of advancing the spread of interest and knowledge in home computing. By bringing together professionals and amateurs, businessmen and women, students, and just plain interested novices, the society acts as a focal point for the distribution of information and help in the field of microcomputing.

RAMS meets on the second Thursday of every month at Rochester Institute of Technology, building 9, room 1030, at 7:30 PM. Meetings usually include a guest lecture on some general interest topic relating to computers, and discussions by members of their own experiences. The "random access" sessions allow anyone with a question to draw upon the help of the entire group. Membership to RAMS costs $5 per year and includes a subscription to the club magazine, *Memory Pages.* For more information, or a free copy of *Memory Pages,* write to RAMS, POB D, Rochester NY 14609, or call Glenn Alexander, president, at (716) 377-0697.

Tulsa Computer Society

The TCS had an exhibit at the Woodland Hills CA Personal Computing Expo, and from the look of the photos in the TCS newsletter, it was a success.

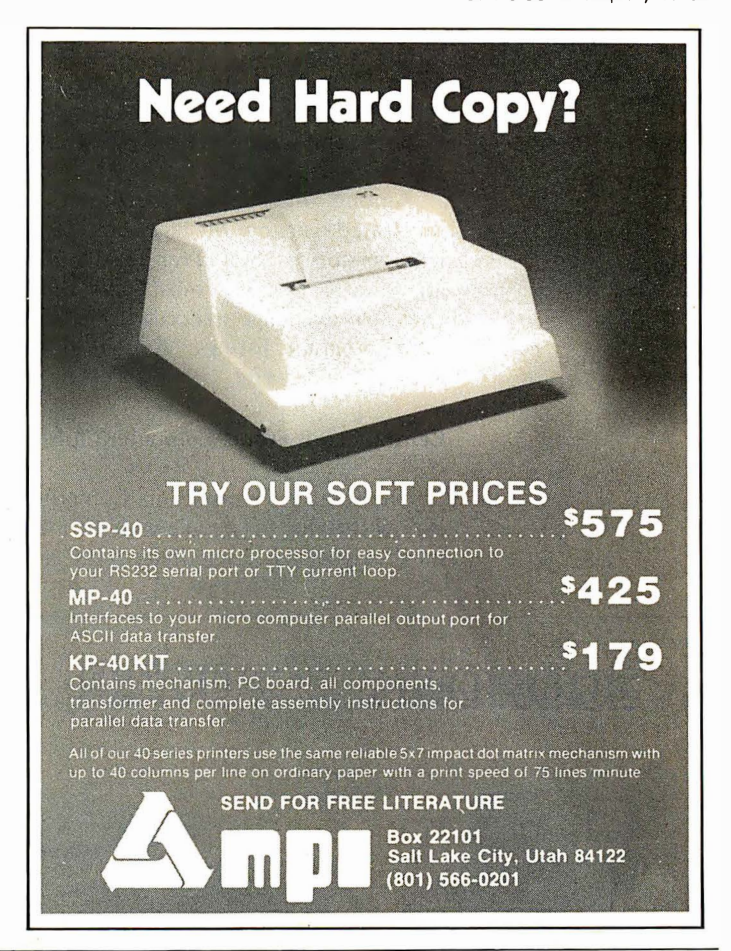
Mike McNatt, editor of *The IO Port* (the newsletter of the TCS) has informed us that they would like to trade the *IO Port* with other newsletters from other clubs. Write to Mike McNatt, c/o TCS, POB 1133, Tulsa OK 74101.

1802 Users' Group in Ontario

Here is another group of hackers who are building their own systems from the ground up using 1802 chips with a kit similar to the Cosmac Elf. The club, numbering approximately 300, is currently working on more memory and IO hardware, and is soon to start on monitor programs and BASIC interpreters for the 1802. Write Tom Crawford, 50 Brentwood Dr, Stony Creek, Ontario CANADA L8G 2W8.

A Compucolor Users' Group

The Compucolor Users' Group is dedicated to the exchange of programs and technical data for the Compucolor color display system. They anticipate issuing a news bulletin periodically. Subjects such as how to concatenate tapes and disks will be covered. For each accepted program a member will receive a number of other programs in return. The initial membership fee of $10 covers the duplication and mailing of materials. Those wishing to join the group may



send the fee to S P Electronics, 5250 Van Nuys Blvd, Van Nuys CA 91401. Further information may be obtained by sending a large self-addressed, stamped envelope to the above address.

Among the present programs are an illustrated version of blackjack, an excellent version of Star Trek, a slot machine, and more. For the most part the group tries to exchange recorded media rather than program listings. Anyone interested is welcome to write.

Central Pennsylvania Computer Club

The Central Pennsylvania Computer Club is now forming for people who are interested in all aspects of computers, both large and small. People in the Philadelphia, Pittsburgh, Baltimore or New York area are invited to contact either Joseph Pallas, 1979 Crooked Oak, Lancaster PA 17601, (717) 569-3137, or David M Ciemiewicz, 533 N Holly St, Elizabethtown PA 17022, (717) 367-6512. They are seeking material for their Newsletter, the *Data Dump*.

Montreal Area Computer Society

Over the past year, the Montreal Computer Society has grown from 12 members to over 90! The club meets once a month, usually on second Tuesday evenings at Vauiev College, 5160 Decarie Blvd. For further information, contact John Erikiev, president, at (514) 932-2344, or write Montreal Area Computer Society, POB 613, Stock Exchange Tower, Montreal, Quebec CANADA.

Space Coast Microcomputer Club

The second edition of the *Space Coast Microcomputer Club Newsletter* has an interesting feature by Paul Rainosek: a tabulated comparison of some of the different microprocessor chips available. The various advantages and disadvantages are clearly listed. Included are the Intel 8080, Motorola M6800, MOS Technology 6502, Fairchild F8, Signetics 2650, and Cosmac 1802. To find out more about the Space Coast Microcomputer Club, contact Ray Lockwood, 1825 Canal St, Merritt Island FL 32952.

Officially LACC

The Louisville Area Computer Club (LACC), formerly Louisville Users of Microprocessors (LUMP), hereby mentions the fact that the club has a new name. So, those

of you who have been looking for "LUMP" meetings should now go to "LACC" meetings. These are held on the first Saturday of each month at 1 PM (usually). Check with this address for the locations: LACC, 115 Edgemont Dr, New Albany IN 47150.

An F8 Users' Club

A group of Fairchild F8 users has started in the Hartford CT area. They have three F8s built and running with two systems having 5 K and 16 K memory, and are interested in contacting other F8 users who would be interested in exchanging information and programs. Contact G W Hemphill, 132 Scott Swamp Rd, Farmington CT 06032.■

The First Annual Micro-Chess Tournament

The first annual Micro-Chess tourney will be held in Louisville KY in August of 1978, sponsored by the Louisville Area Computer Club. To put on a really fair tournament the club is in the process of drawing up the rules and regulations. The preliminary rules are:

- Competition limited to approved 8 bit microprocessors; no bit slice machines will be allowed. Other microprocessors will be considered. Send request with stamped, self-addressed envelope to address below.
- Programs can be in either machine language or a higher level language.
- 16 K 8 bit words memory maximum. (9 bits if parity is used.)
- Homebrew machines and commercial machines allowed.
- Machines may be loaded from any media but after the program is operating the loading device must be detached.
- A panel of judges will rule promptly on program crashes or other unexpected problems.
- Competition will be timed.

The 1979 Micro-competition will allow up to 32 K bytes of memory; the 1980 will allow 64 K bytes (65,536 bytes) of memory. The sponsoring committee reserves the right to alter the rules to meet unforeseen situations. First, second and third place prizes will be awarded. For further information write: Louisville Area Computer Club, 3028 Hunsinger Ln, Louisville, KY 40220.■

Continued from page 22

Listing 1: Most of the lunar landing games I have seen are not flexible enough to run a two degree of freedom real time simulation as described in this article, so I have included this listing. At each initialization, this program finds a random set of starting conditions (speed, position, mass, etc) that is consistent with a safe landing. It then keeps track of speed, position and fuel consumption, printing them as required, and indicating when the surface has been reached. The following adjustments will have to be made by each user:

1. *Function USR(X) must be provided to return the current desired thrust settings, 0 to 100% in the vertical direction, and −100 to +100% in the horizontal direction. These inputs are best achieved by analog to digital conversion from joysticks or slide pots.*
2. *The step size and print interval must be adjusted for your system clock and peripheral speed in order to simulate real time operation accurately.*
3. *Function RND(1.) is assumed by the program to return values between 0. and 1. Alterations may be necessary to suit your version of BASIC.*
4. *The comments printed by the program have deliberately been kept short. A better game could be fashioned by adding instructions, comments on performance, low fuel warning, etc. In other words, customize the simulation to suit your own tastes.*

```
010 REM LUNAR LANDING SIMULATION
020 REM SET FUEL SAFETY FACTOR
025 REM ADJUST TO CONTROL DIFFICULTY
030 LET S=1.3
040 REM SET STEP SIZE AND PRINT INTERVAL
050 LET D=0.01
060 LET K=1.0
070 REM SET GRAVITY ACCELLERATION
080 LET G=1.62
090 RANDOMIZE
100 REM SET NEW STARTING CONDITIONS
110 LET M=1024.+1024.*RND(1.)
115 PRINT "LEM MASS =",M
120 LET F=G*M*(4.+4.*RND(1.))
130 PRINT "MAX THRUST =",F
140 LET A=1.333*F/M-G
150 LET V=F/M*64.*RND(1.)
160 LET U=0.
170 LET Y=V**2./(2.*A)*(1.+RND(1.))
180 LET X=V
182 REM V IS VERTICAL SPEED
184 REM U IS HORIZONTAL SPEED
186 REM Y IS VERTICAL POSITION
188 REM X IS HORIZONTAL POSITION
190 REM HALF OF MASS IF FUEL
192 REM M-P IS FUEL REMAINING
200 LET P=M/2.
210 REM FIND FUEL BURN RATE, I
220 LET I=(2.*Y+V**2./G)/(1.+A/G)
230 LET I=P/(SQR(I/A)*F*S)
240 PRINT "ALTITUDE, SPEED, FUEL, RANGE"
250 REM BEGIN DECENT CALCULATIONS
260 PRINT Y, V, M-P, X
270 LET T=0.
280 IF M=P THEN 360
285 REM GET VERTICAL THRUST
290 LET A=USR(1.)*F/100.
300 REM GET HORIZONTAL THRUST
305 LET B=USR(2.)*F/100.
310 LET M=M-(A+B)*I*D
320 IF M>P THEN 360
330 PRINT "FUEL EXHAUSTED"
340 LET A=0.
342 LET B=0.
350 LET M=P
358 REM PREDICT NEW U, V, X, Y
360 LET V=V+D*(G-A/M)
370 LET U=U+D*B/M
380 LET Y=Y-V*D
390 LET X=X-U*D
395 REM TEST FOR END CONDITIONS
400 IF Y<4. THEN 440
410 T=T+D
420 IF T>K THEN 260
430 GOTO 290
440 PRINT "MODULE HAS LANDED"
450 PRINT "SPEED =",V
460 PRINT "RANGE =",X
470 IF V<8. THEN 500
480 PRINT "BETTER LUCK NEXT TIME"
490 GOTO 110
500 IF X<128. THEN 530
510 PRINT "ITS A LONG WALK TO BASE"
520 GOTO 110
530 PRINT "CONGRATULATIONS, GOOD LANDING"
540 GOTO 110
550 END
```

program should check for end conditions and, if none are found, begin another step.

The speed of a computer makes it possible to find results quickly for times far into the future, even if the step size is quite small. This is fortunate, because as our simulation stands now, an error is introduced at each step that becomes worse as the step size becomes larger. The error occurs because in a real LEM the mass and speed are changing all the time, but in our simulation they can be changed only between steps. A variety of numerical methods have been developed to cope with this problem. In our example, simply using the average of the beginning and ending values in each step would be quite effective. Actually, if the step size is small, say 0.01 seconds, even this is not necessary. It is true that by using the average values the program could be made to run faster, but it would also need to store several extra variables, require more lines of code, and use more memory. Obviously, there are trade-offs to be made among speed, accuracy, complexity and size. In each simulation, the programmer must decide which combination is best.

For our games application, the combination is not critical. A high degree of accuracy is not required, and the program is short enough that memory requirements should not be a problem. The selection of speed, however, presents an opportunity that is unique to the user of a dedicated system. With a little trial and error, it should be possible to find the step size which causes your system to take exactly 1 second to calculate and display the speed and position

1 second into the future. One machine might do 100 steps of 0.01 seconds and then print the speed, etc. Another might do 64 steps of 0.015625 seconds before displaying new results. Still another, with slow peripherals, might output speed and position only once every 2 or 3 seconds. In any case, as long as the simulated data appears at the same time that real data would, your system will be said to be running a real time simulation. A real time lunar lander game gives you exactly the same time to react as would be given a real excursion module pilot.

To help you implement this idea on your own system, a BASIC language program has been included with this article as listing 1. It should be easy to follow, but a few points are worth explaining. At each step the program will need to obtain the thrust settings. This is done through a function called USR. Because systems differ widely, the content of USR is left to you. Some systems will be able to use a register to hold the thrust; others will access memory location; and some may have to query an input port. Also left to the user is the manner in which the thrust settings are updated. Obviously, they cannot be entered at the keyboard for each 0.01 second step. The keyboard could be used via an interrupt routine however. Ideally, you could implement Thomas Buschbach's joystick interface (March 1977 BYTE, page 88) to allow continuous control of thrust in both degrees of freedom. What began as a simple game will now have become a real time lunar landing simulator requiring quick thinking and a good bit of practice to master.

If you use this idea then my article will have succeeded in its purpose of introducing some of the basic concepts of simulation. Techniques like separating the problem into degrees of freedom, determining the effect of each force separately, and stepping the simulation into the future are all fundamental to any prediction of motion. The differences between this lunar lander game and the complex simulations used in the space program lie in the way forces are determined and in the numerical methods used to calculate speed and position. In future articles, other applications for simulation on microcomputers will be discussed as a means for demonstrating some of those advanced techniques. For now, try applying the ideas presented here to create a game of your own.■

Programming Quickies

Pseudorandom Number Generator

Daniel Grieser
4326 Kenny Rd
Columbus OH 43220

The following algorithm for generating a pseudorandom number is based on the "power residue" method described in an *IBM Data Processing Techniques* bulletin. The modification presented here preserves the brevity of the power residue approach and yields 256 8 bit numbers before repeating. Any seed, including zero, can be used prior to initializing the sequence. The algorithm is simply stated: to obtain the next random number, multiply the previous number by 13 and retain only the least significant 8 bits of the product, then add 1 to the product yielding the new random number. In an 8 bit microprocessor this is accomplished by a series of shifts and additions.

Listings 1 and 2 give an 8080 and a 6800 version of this algorithm, requiring 16 and 15 bytes of storage, respectively. Both are simple subprograms which execute straight through without any loops. ■

Listing 1: An 8080 version of the random number routine. This routine uses registers of the processor as temporaries, and places its result in memory location RND. Multiplying by the number 13 is accomplished with shifts and additions without any looping by noting the identity:

$$13xN = Nx2^3 + Nx2^2 + N$$

The power-of-two factors are generated by left shifts.

Address	Hexadecimal Code	Label	Op Code	Commentary
00 00	21 0F 00	ENTRY	LXI H,RND	Point to RNDM # storage.
00 03	7E		MOV A,M	Retrieve last random #.
00 04	87		ADD A	Shift left once.
00 05	87		ADD A	Shift left again.
00 06	4F		MOV C,A	Store briefly.
00 07	86		ADD M	Add unshifted number.
00 08	77		MOV M,A	Store the sum.
00 09	79		MOV A,C	Retrieve the temporary number.
00 0A	87		ADD A	Shift left.
00 0B	86		ADD M	Add the sum again.
00 0C	3C		INR A	Increment the sum.
00 0D	77		MOV M,A	Store the new random number.
00 0E	C9		RET	Return to calling program.
00 0F	XX	RND	BS	Random number storage.

Listing 2: The equivalent routine specified for a 6800 processor. Note that for both the 8080 and 6800 versions, the code is completely position independent so the absolute object code shown can be used without any modifications.

Address	Hexadecimal Code	Label	Op Code	Commentary
00 00	F6 00 0E	ENTRY	LDA B RND	Load B with last RND #.
00 03	17		TBA	Copy B into A.
00 04	58		ASL B	Shift B left.
00 05	58		ASL B	Twice.
00 06	1B		ABA	Add to A.
00 07	58		ASL B	Shift again.
00 08	1B		ABA	Add to A.
00 09	4C		INC A	Increment A.
00 0A	B7 00 0E		STA A RND	Store result as new
00 0D	39		RTS	RNDM # and return.
00 0E	XX	RND	RMB1	Random # storage.



Circle 6 on inquiry card.

GENERAL LEDGER

for 8080

Compatible with:
MITS Basic
&
IMSAI 8080

Price. $300.00

Send to:

Arrow Computer Service
P.O. Box 271
West Jordan, Utah 84084

Or call: 801-268-0130

Circle 14 on inquiry card.

WANTED:
Adventuresome Souls
For New Horizons.

Horizon 1 Kit $1439
Horizon 2 Kit $1799
Cash Discount Prices

We also sell IMSAI, Cromemco, TDL, Vector Graphic, Polymorphic, Seals George Risk Industries and others. Call for Catalog Now.

(315) 637-6208

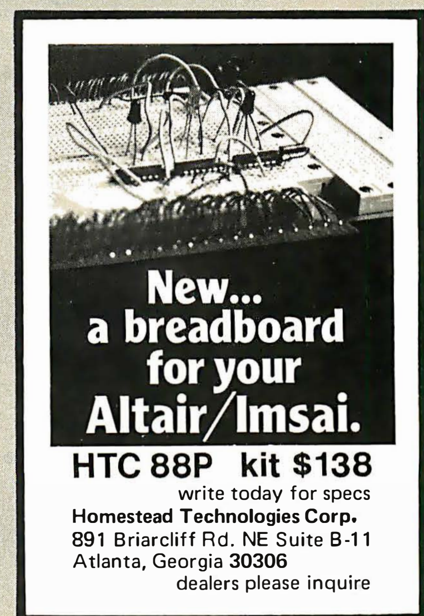
computer enterprises™

P.O. Box 71 • Fayetteville, New York 13066

Circle 29 on inquiry card.



Circle 35 on inquiry card.



HTC 88P kit $138

write today for specs
Homestead Technologies Corp.
891 Briarcliff Rd. NE Suite B-11
Atlanta, Georgia 30306
dealers please inquire

Circle 64 on inquiry card.

Digital Group

System owners now have available the finest Z-80 software ever produced. By an exclusive agreement with TDL we are able to release all their fabulous software ready to run in your DG System.

Here's what you've been waiting for-- complete, user-oriented software! The following programs are available now!

Zapple Basic **$40**
Super Basic **$79**
Macro Assembler **$40**
Text Editor **$30**
Word Processor **$40**
System Monitor **$20**

And soon, a Fortran IV Compiler.

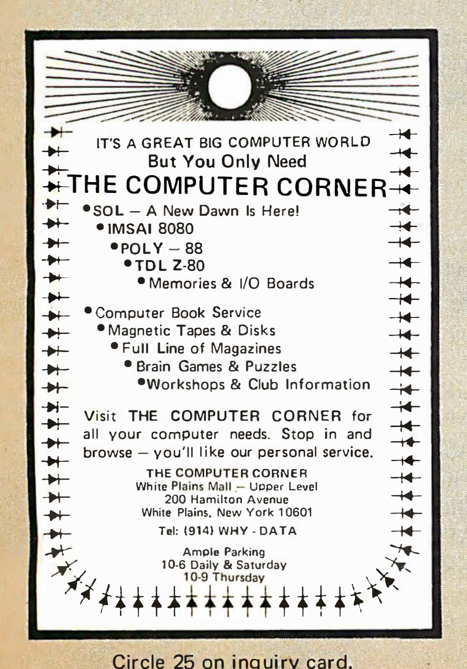
You owe it to yourself to find out. Don't let your hardware be under-utilized. Order now or send for our complete catalog.

Bye Maxi · Hello Super



P.O. Box 4069 Pompano Beach, Fla. 33063

Circle 85 on inquiry card.



Circle 25 on inquiry card.

CANADIANS!

Eliminate the Customs Hassles. Save Money and get Canadian Warranties on IMSAI and S-100 compatible products.

IMSAI 8080 KIT $ 838.00
ASS. $1163.00
(Can. Duty & Fed. Tax Included).
AUTHORIZED DEALER

Send $1.00 for complete IMSAI Catalog.
We will develop complete application systems.
Contact us for further information.

Rotundra Cybernetics



Box 1448, Calgary, Alta. T2P 2H9
Phone (403) 283-8076

Circle 123 on inquiry card.

South Florida

SCS SUNNY COMPUTER STORES, INC.

South Florida's First Computer Store

WE CARRY:
- ALPHA MICRO SYSTEMS, MICRO FILE, IMSAI
- BOOKS, MAGAZINES, NEWSPAPERS
- FULL SERVICE
- SOFTWARE
- WRITE OR PHONE US FOR PRICES

MORE THAN JUST TOYS

Monday 12 noon – 9 pm
Tues.–Fri. 10–6

UNIVERSITY SHOPPING CENTER
1238A SOUTH DIXIE HIGHWAY
CORAL GABLES, FLA. 33146
(305) 661-6042

Circle 139 on inquiry card.

Languages Forum

GRAPLing with APL

William Leler
Visual Comforts Etc
Box 2671
Houston TX 77001.

William Leler sent us the following letter detailing his thoughts about APL and GRAPL. The latter is a new language Mr Leler is currently helping to develop.

I have used APL professionally for many years, implementing such things as Jay Forrester's simulation of the world and a graphics animation package with which I have produced several movies. But there are many severe limitations to APL, some of which become critical on small systems like the typical microcomputer. A few of these are:

- All arrays in APL must be homogeneous, ie: all elements must be of the same type. Since APL does not allow data structures (like PL/I or COBOL), it is sometimes difficult to define variables that are convenient to manipulate. This results in more computer time being spent getting the data into an array for a simple APL matrix multiply than the time required to actually perform the multiply.
- APL simulates operations in parallel so well that there are no constructs for operations serially (such as a looping construct). This leads to much wasted computation (ie: testing all elements of a character vector to see if they are equal to a space when all you really want to do is find the first nonblank character). This may not mean much on, say, the IBM 370, but it slows a microcomputer down.
- APL is terrible for dataset management.
- APL has no interrupt action other than the ability to break to the terminal user. This requires explicit tests to be included for such things as zero divides, etc. There have been patches for this, but they are hard to use.
- Since APL creates and destroys large arrays frequently and at random, storage management must be done with some sort of free space list (which eats up storage) and a garbage collector with full compaction of free space (which really eats up time). This means that, while on a large system most FORTRAN or PL/I programs will completely fit in 128 K, an APL program is given (normally) 64 K bytes just for data and a symbol table, not to mention the space occupied by the APL system.
- Almost no translation can be performed on APL code. Every time a variable name is encountered, it must be looked up in the symbol table. A lot of wasteful data checking must be done before the simplest operation can be performed.
- Character handling in APL is clumsy and difficult to code. Only with great effort can APL code be made self-documenting. Trying to decipher old code (even my own) quickly leads to the funny farm.

But APL has many great features, several of which I wish would be available on a microcomputer. I have found many of these features in a language called GRAPL. GRAPL at first looks similar to APL, but there are changes and improvements, far too many to list here. But just to begin:

- GRAPL uses symbolic operators (like APL), but GRAPL uses the standard ASCII character set with no overstrikes. The large set of operators is derived by allowing two character operators.
- APL execution is from right to left; GRAPL is from left to right. No operators have precedence over others.
- GRAPL is block structured like ALGOL,

which allows a simple and efficient stacked storage management scheme.

- GRAPL programs are just character strings, so, like LISP, code can be written by other programs and executed. This technique is extremely powerful in GRAPL, because much of the GRAPL system is written in GRAPL to allow user modification. For example, in APL, all function editing is done by special function editing routines in a special function definition mode. Editing must be done a line at a time. Thus, simple tasks, such as changing all occurrences of the name A to the name B, are made very time-consuming. In GRAPL, the function editing routines are written in GRAPL so that they are easily modified or rewritten to suit the user's tastes.
- GRAPL has some pattern matching similar to SNOBOL4, which makes tasks such as program editing a lot simpler.
- GRAPL has a nice set of data types ranging from bit strings to integers to real numbers to three-dimensional points and lines. Needless to say, GRAPL is very good at computer graphics.
- As well as looping construct, there is a construct similar to a KEIL structure useful for computer aided instruction and other dialog.
- GRAPL programs are completely free form and, as in the case of FORTRAN, spaces are ignored. This allows programs to be formatted to make reading easier, or allows code to be compressed for efficient storage.
- When a program is executed, it is partially compiled and then executed interpretively. This is one of the fastest methods of execution.
- Errors are handled either by the user or by interrupt routines. Interrupts can be signaled in code or with a timer to allow for such things as concurrent processing and IO, or a limited form of multitasking.
- GRAPL retains the interactiveness of APL.

I have been involved in the formal definition of GRAPL and its implementation on an IBM 370 for a year now, and am beginning a Z-80 version which I hope will fit in 24 K bytes of memory (including some space for user's programs). GRAPL should prove to be not just a pacifier for people who want APL, but a distinct improvement with more general applications. ■

Virtual Memory and VSAM for Micros

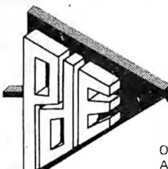
Mark Dahmke
1393 8th St
David City NE 68632

Concerning the APL articles in the August 1977 BYTE, I have not yet seen any mention of direct access file handling for APL. Since many small systems users are processing text, mailing lists, medical records, and scientific data, file searching on a floppy disk would seem to be of great importance.

In the past, the disk or tape access method software was generally cumbersome (and still is in languages like FORTRAN). Unfortunately, I am seeing the same mistakes made in the software development on microcomputers. I suggest taking a look at the virtual storage techniques used on systems like the IBM 370. A special access method called VSAM has been developed that allows data on disk or tape to be treated as if it were in programmable memory. Instead of giving a file record number, or track and cylinder address, one simply gives the address of the particular byte or block to be retrieved, and VSAM does the conversion to physical address. This also makes VSAM device independent! My suggestion is this: instead of adding features to the interpreter (in the form of READ and WRITE commands as in FORTRAN) to handle direct access files, why not make the entire disk surface part of virtual memory. The available space will be the same as if the older direct access methods are used, but this gives the user the opportunity to store large files and data as arrays in memory. Thus one storage method is used and each disk surface can be treated as one large APL (or other language) workspace.

Since reference was made in one article to the difficulties of handling large arrays in user programmable memory, and the need for more than 20 K of memory to hold the interpreter and workspace, the use of a floppy disk as virtual memory could alleviate most of the problem. In fact, I used to work with APL on an IBM 1130 with one disk and only 8 K of core memory and almost all of the workspace was kept on disk. Most of the 8 K not used by the interpreter was used as temporary storage.

I hope that those who write the new APL interpreters will consider what the new technology has to offer before following the old designs.■



Page Digital Electronics
1701 East Orange Grove Blvd
Pasadena, California 91104
Orders under $25 and COD Orders add $2
All others shipped UPS Ppd.
We accept Visa and Mastercharge

Orders under $25 and COD's add $2
All others shipped UPS Ppd.
Visa and Mastercharge accepted

(213) 797-4002 or 797-4007

Lowest Prices on Wire Wrap Supplies!

Precut Wire Wrap Wire

#30 Kynar wire

	100	500	1000	5000
3"	.82	2.60	4.71	4.22
3½"	.86	2.80	5.12	4.55
4"	.90	3.00	5.52	4.88
4½"	.94	3.21	5.93	5.21
5"	.98	3.42	6.34	5.52
5½"	1.02	3.65	6.75	5.86
6"	1.06	3.85	7.16	6.19
6½"	1.15	4.05	7.57	6.52
7"	1.20	4.25	7.98	6.85
7½"	1.25	4.45	8.39	7.18
8"	1.29	4.65	8.80	7.53
8½"	1.32	4.85	9.21	7.84
9"	1.36	5.05	9.62	8.17
9½"	1.40	5.25	10.03	8.50
10"	1.45	5.51	10.44	8.83
Add+1"	.10	.41	.82	.66

Colors: Red, Blue, White, Yellow, Orange, Green, Black. All lengths are overall with 1" strip on each end.

Wire Wrap Tools

BW360
Battery Operated
Hobby Wrap Tool
Auto-Indexing
Extra Bit $2.75
C Batteries 2/$5
26-28 AWG Bit $7.50

$34.95
with Free
Wire Kit 1

WSU30 tool Hand Wrap/Unwrap/Strip $5.50
WSU30M Same but wraps 1½ turns insulation $6.50

Bulk Wire Wrap Wire

	1 roll	5 rolls	10 rolls
50 ft	1.25	1.15	1.00
100 ft	2.00	1.80	1.75
250 ft	4.00	3.50	3.00
1000 ft	12.00	10.00	8.00

Wire Wrap Starter Kits

Kit 1
$6.95
250 3" 100 4½"
250 3½" 100 5"
100 4" 100 6"

Kit 2
$19.95
500 3"
500 3½" 100 5½"
500 4" 100 6"
500 4½" 100 6½"
250 5" 100 7"
1 Roll 250" wire

Kit 3
$8.95
50 3" 1 WSU30
50 4" 1 soft roll
50 5" wire

Interconnect Cables

Ribbon cable connectors for connecting boards to front panels, or board to board.

	Single Ended			Double Ended		
	14 pin	16 pin	24 pin	14 pin	16 pin	24 pin
6"	1.24	1.34	2.05	2.24	2.45	3.73
12"	1.33	1.44	2.24	2.33	2.55	3.92
24"	1.52	1.65	2.63	2.52	2.76	4.31
48"	1.91	2.06	3.40	2.91	3.17	5.08

Wire Wrap Sockets

Gold 3-Level Closed Entry Design

	1-9	10-24	25-99	100-249	250-999	1K-5K
• 8 pin	.41	.38	.35	.31	.29	.27
• 14 pin	.42	.39	.36	.32	.29	.27
• 16 pin	.46	.43	.39	.35	.32	.30
• 18 pin	.63	.58	.54	.47	.44	.41
20 pin	.84	.78	.71	.63	.59	.54
• 22 pin	1.30	1.20	1.10	.95	.90	.84
24 pin	.91	.84	.78	.68	.64	.59
28 pin	1.25	1.15	1.08	.95	.89	.82
40 pin	1.65	1.55	1.42	1.25	1.15	1.09

•End and side stackable sockets

Wire Wrap Boards

				1-4	5-9	10-24	WW Connector
H-PCB-1	4x4½	44	Buses on both sides	4.75	4.50	4.00	3.00
3662	4x6	44	Blank	6.25	5.85	5.50	3.00
3682	4x10	44	Buses on one side	10.50	10.00	9.50	3.00
3682-2	4x6	44	Buses on one side	9.50	9.00	8.25	3.00
4066-4	4x6	72	Buses on both sides	13.50	13.00	12.50	4.00
3719-1	4x6	72	Blank	9.00	8.50	8.00	4.00
3719-4	4x10	72	Blank	11.00	10.50	10.00	4.00
4350	7x9½	80	Buses on both sides	17.50	16.50	16.00	7.00
8800V	10x5.3	100	Buses on both sides	19.50	17.96	16.95	5.00
169P84	8½x17	–	Blank	6.25	5.75	5.25	–

4K RAM BOARD

Assembled and tested. **$89.95**

See kit below.

4K RAM BOARD KIT

450ns Access RAMs
Fully Buffered
Low Power **$79.95**
Static
5V only
4½ x 6 inch board

Buy 4 RAM Board kits at $79.95 each and an 8 slot Mother Board is yours **FREE.**

Includes 8 connectors and card guides.

80 LINE DIGITAL I/O BOARD

—Bus Fully Buffered—
—I/O Includes 4-6820 PIA's
—Compatible With Our Bus
—5V Only

KIT $59.95

2102 — $1.30 Multiples of 25 only.
Low power, 450 n.s. Access and cycle.

MOTHER BOARD

8 SLOT 44 PIN BUS
50 Pin Edge Connector

Mother Board $20.00 ea
Connectors 2.50 ea
Card guides for above $10.00 per set.

MAKE CHECK OR MONEY ORDER PAYABLE TO:
Kathryn Atwood Enterprises
P.O. Box 5203, Orange, CA 92667

Discounts available at OEM quantities. For orders less than $25.00 total, add $1.25 for shipping. California residents add 6% sales tax. Estimated shipping time 2 days ARO with money order. For checks allows 7 days for check to clear.

Classified Ads

FOR SALE: Two Viatron 21s, one with robot printer. Also, SwTPC 6800 with Ct 1024 factory tested. Two credit card readers. One Univac typewriter terminal. Best offer. Greg Ludwig, POB 408, Rice Lake WI 54868. (715) 234-2680.

SWAP: YASEU FTDX570 amateur transceiver in mint condition for Altair or IMSAI Microcomputer. Specify cash differential and write Tom Harmon, R4 Magnolia Dr, Salisbury MD 21801. (301) 742-4742.

WANTED: Software for the Fairchild F8 microcomputer. Lloyd Lawrey, 8926 N Sammy Ln, Kansas City MO 64155.

EDITOR ASSEMBLER: Enhancements for the IMSAI SCS-1. Have package which will add resequencing of the current file on command; printing of the symbol table on command; a new op code, ASC, which will allow an ASCII string constant of up to 30 characters to be defined per line of source code. Package is stored in memory locations 0E50 — 0FFF. No additional memory required. An INTEL checksum paper tape containing object code, a complete source listing, and complete documentation for $7.50 postpaid. James E Bell Jr, 491 Johannah Pl, Lilburn GA 30247.

FOR SALE: A dozen bidirectional 1.8° stepping motors, new, unused, Computer Devices Model 23H-02, $35 each. Requires 0.5, 1 A/15, 25 V power supply. Driving (stepping) circuit can be assembled for less than $10. Easily controlled via a parallel IO port on a microcomputer. Ray Wilhelm, 40 Bear Mountain Rd, Ringwood NJ 07456.

WANTED: BYTE issues 2, 3, 8 and 9. Contact Larry F Tonar, PFC 547969268, B Co 1st Br USAISDSB, POB 921, Fort Devens MA 01433.

COSMAC: Does anyone in the San Francisco CA area have a COSMAC system? If so would you like to start a COSMAC users group or club? Write or call Don Heiden, 1361 Kansas St, San Francisco CA 94107, (415) 285-6966.

FOR SALE: SwTPC 6800 computer with 16 K memory $600; AC-30 cassette interface $60; and a Lear Siegler ADM-3 dumb terminal $700. All components of this system are burned in and working. I will also include 4 K and 8 K BASIC, Star Trek, and other software free. John Victor, 11 Idar Ct, Greenwich CT 06830, (203) 869-4479.

FOR SALE: New SwTPC CT-1024 video terminal, KBD-5 keyboard option, CT-CA cursor option, CT-S serial option, CT-P power supply, in custom black metal case, running and ready to plug into your SwTPC 6800, $450. Reason for selling, using HAL DS-3000 for both ham station and computer. K Slobb, W9BT, 1605 Oakwood Rd, Northbrook IL 60062, (312) 272-1223.

Readers who have equipment, software or other items to buy, sell or swap should send in a clearly typed notice to that effect. To be considered for publication, an advertisement should be clearly noncommercial, typed double spaced on plain white paper, and include complete name and address information. These notices are free of charge and will be printed one time only on a space available basis. Insertions should be limited to 100 words or less. Notices can be accepted from individuals or bona fide computer users clubs only. We can engage in no correspondence on these and your confirmation of placement is appearance in an issue of BYTE.

Please note that it may take three or four months for an ad to appear in the magazine. ■

FOR SALE: HP9815A programmable calculator with extended memory, carry case, four cartridges, thermal paper, manuals; excellent condition, over $3600 new. Asking S2600. Walt Goldys 200 S Glenn Dr, =53G, Camarillo CA 93010. (805) 482-4674.

FOR SALE: 113 point EDP physical security checklist with risk management briefing and bibliography. $5 postpaid. D J Scherer, 29 Reymont Av, Rye NY 10580.

FOR SALE: ASCIISCOPE 12 by 80 $550 with documentation; Solid State Music MB3 with 4 K 1702A $125; MITS 4 K Dynamic $125; Proc Tech VDM $170; 3P and S $135; Pennywhistle MODEM $80; OAE Tape Reader $65; IMSAI 4 K STATIC $120. All assembled fully socketed and factory checked out. K R Roberts, 10560 Main St, Suite 515, Fairfax VA 22030.

FOR SALE: One IBM Model 735 IO Selectric typewriter. Correspondence wiring. 15 1/2 inch platten and prints 132 columns. Good documentation. Good condition. $650. Sam Weiman, 9396 Fernbury, Cypress CA 90630. (714) 827-7432.

FOR SALE: New Sphere computer system, still in unopened boxes from factory. System contains: 24 by 80 chr CRT, 36 K programmable memory, 2 serial interface ports (one cassette), 4 parallel interface ports, keyboard, power supply, dual Orbis floppy disk drive, and an Okidata line printer with tractor feed. Price $6850. T W Reeves, 290 Almak Ct NW, Issaquah WA 98027. (206) 392-1447 or (206) 655-5308.

FOR SALE: Complete set BYTEs, 1 to present, $75. John Wagner, 2175 Wagner Dr, Caro MI 48723.

WILL SELL: 15 cps Selectric IO printer, 1621 paper tape punch and reader, IBM console keyboard printer, for best offers; IBM 1311-3 2 M drive with two disk packs, $400 or best offer. All stuff with full documentation. Henry Birdseye, 316 Jefferson NE, Albuquerque NM 87108. (505) 268-4727 10 PM to 11 AM.

FOR SALE: IMSAI PI04-4 parallel and SIO2-2 serial interface boards. $155 each, cables also available. Various 4 K programmable memory boards, $125 each, all assembled and tested. BYTE numbers 1 thru 4, $10. Trade any boards toward floppy disk? Dieter Kaetel, .7201-87 SE, Mercer Island WA 98040. (206) 232-1513.

FOR SALE: Pertec 7850 9 track magnetic tape drives with elec, $275 with manual; ±12 V 3A and ±5 V 5A supplies, $25; Superior Elec TRP125 120 bps optical reader, $200; IVC 600 color video recorders with manual $300. (originally $2700) with tape; MSI FD8 disk case and power supply $85. Gary Gaugler, 2276 Beaver Valy Rd, Fairborn OH 45324. (513) 878-0288.

FOR SALE: Altair 8800a processor $600, 88 4MCS 4 K static memory card $200, 88-16MCS 16 K static memory card $800, 88-DCDD disk drive and controller $1500, 88-2SIO serial Teletype interface $150, 88-ACR audio cassette interface $150. All units factory assembled, tested, brand new. Factory new ASR-33 Teletype $1000. Michael Clark, RD 3, Nazareth PA 18064. (215) 759-6873.

TRADE: First 24 issues BYTE magazine for RO/ASR 33 in good condition. Alvin L Hooper, 207 Self St, Warner Robins GA 31093. (912) 923-5235.

FOR SALE: Card reader, 100 cpm optical NCR EM-D2, also with spare parts, $75. Arthur Okun, 1803 N Mulligan Av, Chicago IL 60639. (312) 637-0938.

FOR SALE: Teletype Model ASR-33 with MITS call-control unit Model 88-TYA, stand and one case of roll paper, brand new, never used, $995 or best offer. Bob Majdanski, 214 Coolidge Av, Hasbrouck Heights NJ 07604. (201) 288-3742 after 7 PM.

FOR SALE OR TRADE: Texas Instruments LCM-1001 Microprogrammer trainer with manual and book, *Software Design for Microprocessors*, all in excellent condition. $100 or in trade towards HP-25, 55, 65 or amateur radio gear. Also have complete file of BYTE in mint condition for sale or trade. David J Lowenstein, 235 E 15th St, Tempe AZ 85281. (602) 966-0140.

FOR SALE: New (three months old!) IMSAI 8080 microcomputer system. This complete and functional system includes: Seals 8 K memory, Polymorphic video display board connected to GBC monitor and SwTPC keyboard. Tarbell cassette interface attached to GE cassette recorder, assembler and BASIC on tape, all manuals, monitor, keyboard, and recorder disconnect at back panel for greater mobility. Perfect for hobbyist, $2350. Joel Schwartz, 5 The Maples, Roslyn Estates NY 11576. (516) 484-5732.

FOR SALE: 3M DC300A data cartridges which work in the IBM 5100, TEKRONIX, DEC and similar equipment at a club price of $18 per data cartridge plus the cost of postage. IBM 5100 Users Group, c/o HITS Inc, 5541 Parliament Dr, Suite 104, Virginia Beach VA 23462. (804) 490-0154.

FOR SALE: Complete NCR 315-100 data processing system. Includes 315-100 mainframe, console with 1933 Teleprinter and power supplies, 10 K by 12 bit core memory, five 334 7 track ½ inch magnetic tape drives, 340-503 high speed printer (120 characters per line, 800 lines per minute) and an IBM 1442 card reader/card punch. All units are intact and in perfect operating condition. Complete schematics, service and operating manuals are included. Looking for offers on all or any part. Gary Boehm, 1671 Timmy Dr, Hamilton OH 45011.

FOR SALE: HP9810A computer system including 10 by 15 inch plotter; audio cassette data storage unit; ROMs for plotter, cassette, math functions, alphanumeric printing, definable functions; plus lots of general-purpose software written by HP and me. This system is ideal for a scientist or engineer doing independent research who cannot afford a full-scale computer system, but who needs good quality plots for publication in journals. West Los Angeles area. Call Gary Bedrosian weekdays at (213) 478-3035.

FOR SALE: BYTE #1, 2 and 3, $25; #9 and 10, $15; all five, $30. Wanted: Used, working microcomputer system under $1000. Donald Erickson, 6059 Essex St, Riverside CA 92504, (714) 687-5910 2 PM to 9 PM any day.

FOR SALE: Viatron Print Robot and plans for interface, $60. Converts Selectric and other typewriters to printers without internal mods, via keyboard. Like new, G Lyons, 280 Henderson St, Jersey City NJ 07302, (201) 451-2905.

FOR SALE: Six 4 K dynamic memory boards (MITS), $150 each, and two 4 K static boards, $150 each. Robert K Snider MD, Medical Arts Center, 1230 N 30th St, Billings MT 59101.

FOR SALE: 6800 microcomputer system: CPU board with 1 K EPROM (monitor, debug, assembler, editor), 4 K RAM, ½PIA; 16 K RAM memory board; video interface (16 by 32); serial IO with two ACIA ports (audio KC standard and baud select RS232), EPROM (load and dump), and PC wiring for TTY/TTL/modem; 9 inch TV monitor; cassette recorder; manuals, schematics, documentation; software on cassettes includes Tiny BASIC, games, and more. Price: $1800. Contact David Domorest, 12697 Graton Rd, Sebastopol CA 95472, (707) 823-1698. Factory assembled and tested.

FOR SALE: Sphere 330 Computer System. Includes: 6800 CPU, 20 K memory, 32 X 16 video interface, ASCII keyboard and interface, dual cassette interface, case less cover, power supply, manuals, monitor, editor, and debugger routines in ROM including cassette IO, 16 bit math, etc. Will also supply Tiny BASIC and 16 K BASIC. Complete system as a kit originally cost over $1700. This is an assembled, tested, burned-in system for only $1500. First check to clear takes it all. Write Richard H Rae, POB 791, Emporia VA 23847.

WANTED: Listing or source of listing of assembler program for 8080 (home-brewed RM-8080), for reasonable price. T P Douglas, POB 1012-C, La Junta CO 81050.

FOR SALE: VIATRON microprocessor system. Includes keyboard, two cassette transports, video monitor, and power supply. Only $400. Write or call Mike Vitale, POB 22, Suncook NH 03275. (603) 485-4006.

FOR SALE: Frieden Selectradata paper tape reader and card reader $50 each. R Rodgers, 11 Skip Rd, Norristown PA 19403. (215) 279-5761.

WANTED: Medical software, also artificial intelligence and robotics. Send description and price or trade. Rob Lufkin, UVA School of Medicine, Charlottesville VA 22901.

FOR SALE: Altair 8800, 4 K static, 4 K dynamic, Serial IO (RS232 compatible), Tarbell cassette interface, complete documentation, new cost $1576, barely used and totally functional, $1375 or best offer. Norm, 8508 Lurline Av, Canoga Park CA 91306. (213) 341-1275.

FOR SALE: First 24 BYTEs, all like new. Best offer. J L Cutright, 1417 N Hoyt, Chillicothe IL 61523.

F-8 USERS: Have designed a 4-1/2 inch by 6-1/2 inch double sided PC card with plated thru holes for an F-8 system consisting of a 3850, 3853, up to two 3851 or 3861 chips as well as space for two 2708 or 2716 EPROMS. For dedicated applications where no programmable memory is needed. Need to sell ten at $250 each to recover costs of development and prototypes. Homer S White, 3314 Pickett Rd, Durham NC 27705.

FOR SALE: QUME Sprint 55 printer, equivalent to Diablo printer but better and faster. Has been modified by factory to use Xerox print wheels, including proportional spacing wheels, new. I paid $2950. Please bid. Puran, POB 135, Jamaica Plain MA 02130.

FOR SALE: Video checkers on Tarbell compatible cassette. Plays under MITS 8 K BASIC. Total of 16 K memory and Poly Video board required. Send $10 postpaid for cassette and complete documentation to Marv Mallon, 6914 Berquist Av, Canoga Park CA 91307.

FOR SALE: IMSAI 8080 with 22 slot mother board $649, Polymorphics video board VTI-64 $199, Vector Graphics 8 K memory board $235, Solid State Music 8 K memory board $225, Radio Shack keyboard with custom case $49, and a Tarbell cassette interface $109, assembled, tested, in excellent condition. B Marr, 1800 Brea Blvd, Box 18, Fullerton CA 92635. (714) 870-1387.

FOR SALE: Line printer, 300 cpm 132 column drum printer. Operational Mohawk Data Sciences series 4000 has full documentation and 63 printable character set. Best offer over $800, FOB Madison WI. Richard Whitnable, 425 Sidney St, Madison WI 53703. (608) 256-3789.

WILL TRADE: November 1975 BYTE for December 1975 BYTE. My November issue is in very good condition. Elliot S Wheeler, 101 Volney St, So Houston TX 77587.

WANTED: Two 1 K memory boards for Mark 8 microcomputer (from July 1975 *Radio-Electronics*) as produced by Techniques Inc, prefer unpopulated boards. Jeff Lesinski, 1241 Staley Rd, Grand Island NY 14072. (716) 773-3783.

WANTED: Old PDS 1020E paper tape software. Can copy and return. Also need any circuit board for connectors. Dave Overton, 1709 W 30, Austin TX 78703.

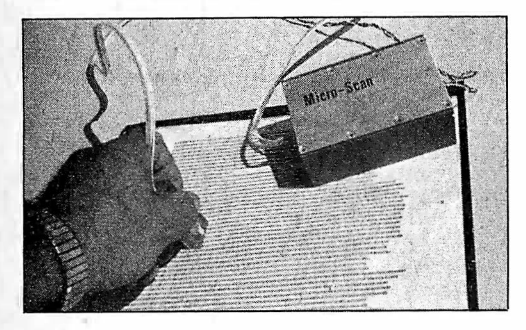
FOR SALE: QUAY 80A1 do everything processor, a complete Z80 stand alone computer. Add Teletype or RS 232 terminal and you're up and running with 1 K static programmable memory (low power), 512 byte (ROM) monitor, 4 UVEPROM sockets (2708), UVEPROM programmer, and more, assembled and tested $425. Also TDL ZPU with software (8 K BASIC, 2 K monitor, text editor, macro assembler) all on original paper tape with documentation, assembled and tested $225. MFE Model 250 super cassette, high speed drive (40 ips) new with case, connector and much documentation. Cost $510, your cost $225. Paul Lamar, 1024 17th St, Hermosa Beach CA 90254. (213) 374-1673.

FOR SALE: Altair 8800 Computer, with 20 K of fast (no wait state) programmable memory, 1 K of slow (1 wait state) programmable memory, 2 K of ROM, Processor Technology video display module, and 3 parallel and serial interface cards. MITS audio cassette interface card. Price: $1500 firm. I will pay shipping. M Harris, POB 1053, New Britain CT 06050. (203) 225-0504.

FOR SALE: Two MITS 88-S4K (4 K synchronous) memory boards, $110 each. Two MITS 88-4MCD (4 K dynamic) memory boards, $90 each. All boards assembled with sockets for each integrated circuit. Every bit OK. A spare 4 K memory chip is included if you buy all four boards. Harold Corbin, 11704 Ibsen Dr, Rockville MD 20852. (301) 881-7571.

What's New?

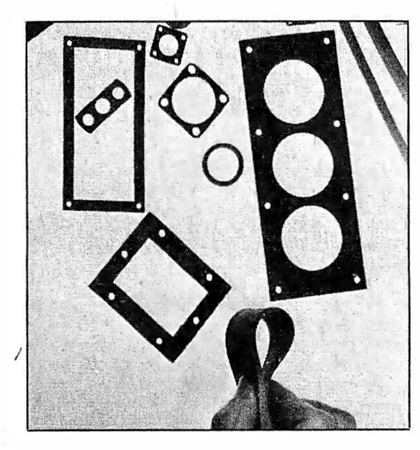
A New Bar Code Scanner



Micro-Scan Corporation has announced a new bar code scanner designed to read bar code programs as featured in BYTE. The unit, called the Micro Scan, is capable of scanning varying contrast ratios (photostatic copies can be read) at rates of 10 to 36 inches (25 to 91 cm) per second. Documentation of loader programs for the 6800, 6502, 8080 and Z-80 processors are provided with each scanner. Power supply requirements are 11.5 to 18 V, unregulated. The Micro Scan is available for $97.50 from Micro Scan Corporation, POB 705, Natick MA 01760, (617) 655-5406.■

Circle 517 on inquiry card.

Shield and Seal for Real



A new conductive material with excellent sealing properties has been announced by Radcon Corporation, 246 Columbus Av, Roselle NJ 07203, (201) 241-5550. Free samples as well as a data sheet giving compression and electrical characteristics are available upon request. The material, called Multi-Con 2, is priced based on customers' applications.■

Circle 518 on inquiry card.

An Unregulated DC Power Supply for Microprocessor(s)

A new DC power supply which provides unregulated power to microprocessors and peripheral equipment has been announced by Standard Power Inc.

Designated the SMP-30B, the unit provides three voltages of 9 VDC at 1 A and ±18 VDC at 0.5 A. It may be operated at 115 or 230 VAC, 50 or 60 Hz input.

Priced at $27.50 (single quantity), the unit measures 3 3/8 by 3 3/8 by 4 3/4 inches (8.57 by 8.57 by 12.1 cm) and weighs 2.1 pounds (0.95 kg).

Details are contained in Standard's Catalog C477, available on request from local distributors, or from Standard Power Inc, 1400 S Village Way, Santa Ana CA 92705, (714) 558-1172.■

Circle 519 on inquiry card.

Attention APL Lovers. . .

MCM Computers is a Canadian firm which has been marketing small desk top APL machines for about five years. As this issue was going to press, we received word that the company is making available a $5000 package consisting of a complete self-contained computer with APL interpreter and dual cassette tape drives for work space. Contact the US office, 2125 Center Av, Fort Lee NJ 07048, (201) 944-2737.■

Circle 520 on inquiry card.



Altair Offers Microcomputer Timesharing



Anyone who has an Altair 8800 series computer can now convert it to serve as the control center for a timesharing system. A special version of BASIC, called Timesharing BASIC, has been developed along with Altair Timesharing Disk BASIC. Both are extensions of Altair extended BASIC and allow up to eight independent programs to be run simultaneously.

A memory partition technique is used to keep each program job in a unique area of memory. Each program area contains the BASIC program text, variable and string space, a workspace, plus approximately 300 bytes of the timesharing system. The system can be used with a variety of IO devices including video displays and printers.

Control of a specific job may be transferred from one terminal to another with a single command. Various control characters allow suspension and resumption of each job without loss of data. Diagnostics are provided for program debugging and automatic line numbering is available during program entry. Both versions of Altair Timesharing BASIC furnish line oriented text editor with line and character manipulation capabilities.

Extensive hardware is needed (in addition to the 8800 series mainframe and processor) to support both versions of Altair Timesharing BASIC. This includes a minimum of 32 K bytes of programmable memory, a vectored interrupt real time clock card, up to four 2SIO serial interface boards for terminals and an optional line printer for the disk BASIC version. The disk version, of course, requires a floppy disk peripheral.

Contact MITS, 2450 Alamo SE, Albuquerque NM 87106.■

Circle 451 on inquiry card.

Attention Toronto Readers

Computer Mart in Toronto has been in operation since January 1977 and maintains at least three systems up and running for demonstration purposes. The store offers complete service facilities as well as programming services for microprocessor based systems. This includes operating system enhancements and accommodating unusual interface situations, both software and hardware. The store's product line includes Processor Technology, Polymorphic, The Digital Group, Peripheral Vision, Cromemco, IMSAI, iCOM, IASIS, Lear-Siegler, North Star, TSC Software, Hitachi and Sanyo, Volker-Craig, Scientific Research, Sams, Hayden and AP Products, etc. Computer Mart's policy is to provide the most effective guidance and general advice to our customers and continue this policy after the system is plugged in at the customer's home. The store address is 1543 Bayview Av, Toronto, Ontario M4G 3B5 CANADA, (416) 484-9708.■

Circle 452 on inquiry card.

A Small Shank Electric Drill from Wahl



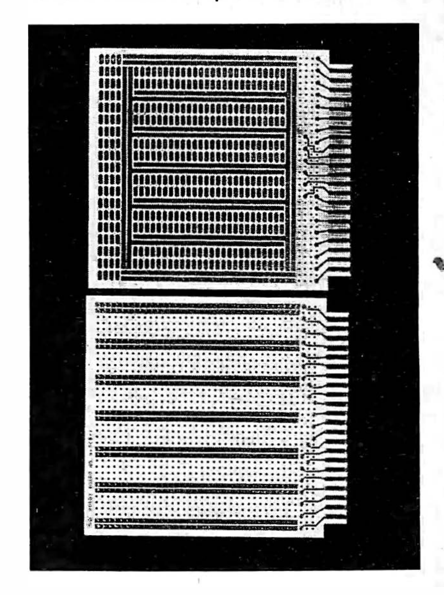
Here is a device that should prove useful for the fabrication of printed circuit boards, as well as other applications involving the drilling and cleaning of small holes. The Wahl ISO-TIP electric drill is less than 5 inches (12.7 cm) long with drill bit removed and is designed to fit into tight corners. The on-off switch provides both intermittent and locked modes of operation, and the power cord is 10 feet (3.04 meters) long.

Operating at 9000 rpm, the drill is supplied with a collet chuck, three collets and two drill bits (#56 and #71). The unit is available in either 110 VAC or 12 VDC versions.

Contact the Wahl Clipper Corporation, 2902 Locust St, Sterling IL 61081 (815) 625-6525.■

Circle 453 on inquiry card.

A New General Purpose PC Board



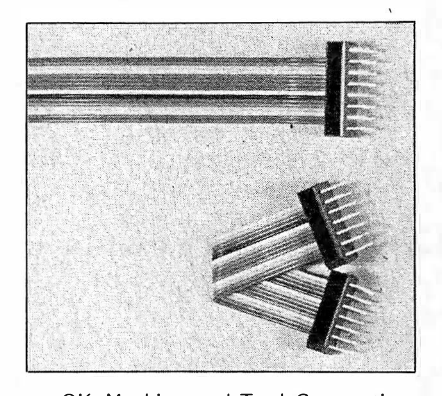
Model H-PCB-1 is the first in a series of PC boards. The 4 by 4.5 by 1/16 inch (10.16 by 11.43 by .19 cm) board is made of glass coated epoxy laminate and features solder coated 1 oz copper pads and has a 22/22 two sided edge connector.

The board contains a matrix of .040 inch (.1 cm) diameter holes on .100 inch (.25 cm) centers. Two independent bus systems are provided for voltage and ground on both sides of the board. In addition, the component side contains 14 individual buses running the full length of the board which enable direct access from edge contacts to distant components.

Priced at $4.99 from OK Machine and Tool Corporation, 3455 Conner St, Bronx NY 10475, (212) 994-6600.■

Circle 454 on inquiry card.

Data On the Cable?



OK Machine and Tool Corporation, 3455 Conner St, Bronx NY 10475, (212) 994-6600, has sent a picture of the new dual in line package cable termination assemblies, retailing at $3.75 to $4.35. Variations on this theme include double ended cables in lengths of 2, 4 and 8 inches (5, 10 and 20 cm) and single ended cables in lengths of 12 and 24 inches (30 and 61 cm); either 14 or 16 pin cables are available.■

Circle 455 on inquiry card.

California Industrial

Post Office Box 3097 B • Torrance, California 90503



CALCULATOR WRIST WATCH

Designed for the on the go executive, that individual who has to make those on the spot decisions.

Handsome gold tone stainless steel watch features space age micro-circuitry. The MOS integrated circuit contains the equivalent of more than 10,000 transistors.

This LED wrist watch displays date, time, elapse seconds and also functions as an eight digit calculator with memory. Information stored in memory can be recalled at any later date, even weeks or months. Use this memory feature to store phone numbers, parking stall location or flight departure time.

Manufactured by one of California's leading aerospace contractors. Because of the discount price we have agreed not to publish the manufacturer's name.

Includes batteries, jewelry case and 18-month factory warranty.

$149.50

ELECTRONIC ENTERTAINMENT CENTER

Tennis·Handball Hockey·Smash

Action-packed color entertainment for the whole family. Adjustable skill level controls allow players of all ages to compete in tennis, hockey and handball. This four game entertainment center turns your television into a video playground.

On screen scoring, live action sound and true component color makes this video center an excellent buy at only $24.88. Complete with antenna box and AC adapter.

Color $24.88

TELETYPE® MODEL 43

New from Teletype, the Model 43 is capable of printing 132 ASCII characters per line. Send and receive data at 10 or 30 words per second. Keyboard generates all 128 ASCII code combinations. RS-232 interface, same as the popular Model 33. Data sheet sent upon request. Manufacturer suggested price $1377.00.

IMMEDIATE DELIVERY $1199

We also have for sale a limited quantity of used Model 35's. Priced at only $449.50

HEXADECIMAL KEYBOARD

$29.95

Maxi-Switch hexadecimal keyboards are designed for microcomputer systems that require 4-bit output in standard hex code.

Each assembly consists of 16 hermetically sealed reed switches and TTL "one shot" debounce circuitry.

Reliable low friction acetal resin plungers are credited for the smooth operation and long life of this premium keyboard.

Requires single + 5 volt supply.

DIGITAL ALARM CLOCK

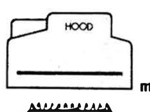
Completely Assembled $19.95





Walnut-grained decorator clock features large .7" LED display which is driven by the new National MM5385 alarm clock chip. Preset 24-hour alarm function allows you to awaken at the same time each morning without resetting. Upon reaching the wake-up time, the clock's loudspeaker emits a gentle tone. Touch the snooze button and doze off for an additional 9 minutes of sleep. Clock also functions as a ten-minute elapse timer. "Alarm Set" indicator, AM-PM display.

CONNECTORS

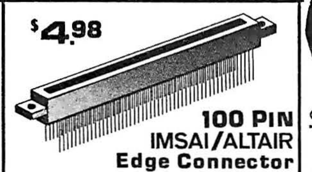


RS-232

DB25P male plug & hood **$3.95**

DB255 female **$3.95**

$4.98



100 PIN IMSAI/ALTAIR Edge Connector

Altair, Imsai compatible gold plated, dual 50 (.125 centers) three tier wire wrap edge connector. 3 for $13.50



$24.88

SPERRY UNIVAC KEYBOARD

The famous Sperry Univac 1710 Hollerith keyboard assembly is now available from California Industrial for only $24.88. The ideal computer input device for accountants and mathematicians. The numeric keys are placed on the lower three rows to resemble a ten key adding machine. This format allows one handed numeric data entry.

Original cost was $385. Used but guaranteed in excellent condition. Complete with documentation.

Quiet Buss™ S-100 MOTHER

The Quiet Buss from California Industrial is quality engineered. No short cuts have been taken to produce this mother board. Active termination circuitry prevents noise and crosstalk. Manufactured from extra heavy FR-4 epoxy glass. Features 2 ounce double thickness copper traces. Purchase this board along with the power supply kit below and you have the start of a super Micro-System.

10 SLOT 24.88 18 SLOT 29.95

FREE

MANUAL GRAPHITE DISPLAY GENERATOR

Modern technology has pioneered the development of this unique character printer. Our Manual Graphite Display Generator has the capability of producing the full upper and lower case ASCII set. Self-contained cursor assembly allows the operator to eliminate erroneously entered information.

Each unit is manufactured to strict tolerances as prescribed by standards set forth by California Industrial. One free with every order.

$4.98 10 for $45.

Scotch® BRAND DISKETTES

Please specify IBM 3740 series or 32 sector.

Certified Digital CASSETTES

Won't drop a BIT!

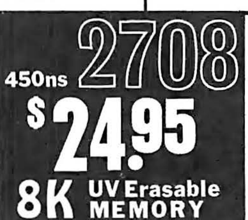


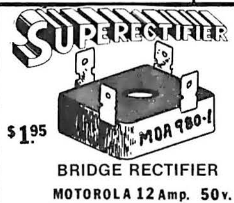
$4.98

JOYSTICK $4.50



This joystick feature four 100K potentiometers, that vary resistance proportional to the angle of the stick. Perfect for television games, quad stereo and radio controlled aircraft.





$1.95

BRIDGE RECTIFIER

MOTOROLA 12 Amp. 50v.

the GEORGE RISK ASCII KEYBOARD KIT

GRI

Model 753k **$59.95**



LOW POWER 450 NS



Lowest Price Anywhere

Our low power static RAMs are factory prime. Purchased, on contract, directly from one of California's leading semiconductor manufacturers.

Access time guaranteed faster than 450nS. Minimum purchase 32 pieces.

Sorry, credit cards can not be accepted on 21L02's.

7400	.13
7401	.19
7402	.19
7403	.19
7403	.19
7404	.19
7405	.19
7406	.19
7407	.25
7408	.25
7409	.25
7410	.19
7411	.25
7412	.35
7413	.49
7414	.79
7416	.39
7417	.39
7420	.19
7422	.49
7423	.39
7425	.39
7426	.39
7427	.39
7428	.49
7429	.39
7430	.25
7433	.39
7437	.39
7438	.39
7439	.39
7440	.39
7442	.59
7443	.79
7445	.89
7446	.99
7447	.99
7448	.99
7450	.25
7451	.25
7453	.25
7454	.25
7460	.25
7470	.25
7472	.39
7473	.39
7474	.35
7475	.49
7476	.39
7479	3.99
7480	.79
7482	.99
7483	.99
7485	.99
7486	.49
7488	3.40
7489	2.79
7490	.49
7491	.99
7492	.49
7493	.49
7494	.79
7495	.79
7496	.79
7497	3.99
74100	1.19
74107	.39
74109	.49
74110	1.79
74116	1.99
74120	1.79
74121	.39
74122	.39
74123	.69
74125	.59
74126	.59
74128	.49
74132	.99
74136	.89
74141	.99
74145	.99
74147	2.49
74148	1.99
74150	1.19
74151	.99
74153	.89
74154	.99
74155	.99
74156	1.29
74157	.99
74159	2.99
74160	1.19
74161	.99
74162	1.49
74163	.99
74166	1.19
74167	4.99
74170	2.49
74173	1.49
74174	1.19
74175	.99
74176	.99
74177	.99
74192	.99
74193	.99

CMOS

4001	.25
4002	.25
4006	1.99
4007	.25
4008	1.49
4009	.69
4010	.69
4011	.25
4012	.25
4013	.49
4014	1.49
4015	1.39
4016	.69
4017	1.29
4018	1.69
4019	1.79
4020	1.39
4021	1.49
4022	1.25
4023	.25
4024	1.19
4025	.25
4027	.69
4028	1.25
4029	1.99
4030	.69
4032	.49
4033	1.99
4035	1.69
4040	1.99
4041	1.49
4042	1.49
4043	1.49
4044	1.49
4046	2.49
4047	2.49
4049	.79
4050	.79
4051	1.99

CPU's

8080A	14.95
6800	24.95
Z80	39.95

MEMORY

1702	5.95
82s23	2.95
82s123	2.95
2102	1.79
2102-1	1.89
21L02	1.19

CLOCK's

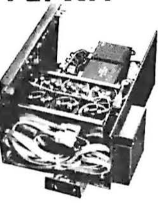
5314	2.95
316	4.95
5375	3.95

LM741 $.09

300H	.79
301H	.39
301CN	.39
302H	1.29
304H	1.29
305H	.99
307H	.49
307CN	.39
308H	.99
308CN	.99
309H	1.09
309K	.99
310H	1.19
310CN	1.19
311H	.99
311N	.99
312H	1.99
318H	1.79
318CN	1.49
319CN	1.29
320K-5	1.39
320K-12	1.39
320K-15	1.39
320T-5	1.75
320T-8	1.75
320T-12	1.75
320T-18	1.75
320-24	1.75
324N	1.79
339N	1.69
340K-5	1.95
340K-8	1.95
340K-12	1.95
340K-15	1.95
340K-18	1.95
340K-24	1.95
340T-5	1.79
340T-12	1.79
340T-15	1.79
340T-24	1.79
350N	.99
351CN	.65
370H	1.29
370N	1.29
373N	3.19
377N	3.99
80N	1.39
381N	1.79
382N	1.79
NE 55v	.49
NE556	1.29
NE565H	1.49
NE565N	1.79
NE566N	1.25
703CN	.45
709H	.39
709N	.39
710N	.79
711H	.39
711N	.39
723H	.55
723N	.55
725H	3.49
733H	1.49
733N	.99
739N	1.19
741N	.09
747N	.79
748H	.39
748N	.39
1414N	1.75
1458	.69
1496N	.95
2111N	1.95
2901N	2.95
3065N	.69
3900N	.49
3909N	1.19

MICROCOMPUTER POWER SUPPLY KIT





Ideal supply for S-100 systems. Contains all the necessary components to produce 5 volts @ 10 A., ± 12 @ 2.5 A. (voltage actually higher). New surplus from Memorex Corporation.

Kit includes 117 ac. transformer, diodes, six computer grade electrolytics, chassis, circuit breaker and grounded power cord. Transformer has 5 output windings.

Include $9.00 shipping East of the Mississippi; $4.00 for Calif.; all other States $7.00.

$24.88 Plus Shipping

SOLID-STATE MICRO BUZZER

$.98

1.5 TO 3 VDC

Thumbwheel switch

Ten position BCD

$1.39 ea.

10	50
$1.19	.89

EECO

SPDT MINIATURE TOGGLE SWITCH

$.98 ea.

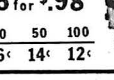
CK

10	50	100	1k
$.88	.81	.73	.66

TRIMMER POTENTIOMETERS

2K 10K 50K

5 for $.98



20	50	100
16¢	14¢	12¢

CAPACITORS

ELECTROLYTICS

	ea.	10	50
4500/50v.	$1.49	1.35	1.19
1000/15v. axial	$.55	.49	.45
.1 disc	$.12	.09	.07
.01 disc	.06	.05	.04



$.98 ea.

DPDT ROCKER SWITCH

32 Conductor RIBBON WIRE

$.59 Ft.

SPECTRA-STRIP

COMPUCORP Power Adapter

shipping $.75

output 7vdc 1.4A

$4.98

RELAYS

SPDT MINIATURE

$1.39 ea.

10	25	100
$1.15	1.04	.89

Coil 5 or 12 Volt dc.

3 Amp Contacts

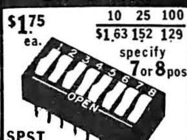
P. C. Board Mount

DIP Switch

$1.75 ea.

10	25	100
$1.63	1.52	1.29

specify 7 or 8 pos.



SPST

Transistors

	ea.	10	50	100
2N2222A	.20	.18	.16	.15
2N3055	.89	.84	.77	.65
MJ3055	.99	.94	.87	.75
2N3772	2.19	1.95	1.75	1.59
2N3904	.15	.11	.09	.07
2N3906	.15	.11	.09	.07

Diodes

		10	25	100
1N4002	100v.	.08	.06	.05
1N4005	600v.	.10	.08	.07
1N4148	signal	.07	.05	.04

jumbo red LED's

ea.	10	25	100
$.15	.13	.11	.09

$.98

Output: 12v. 175ma.

TRANSFORMER

9 foot

$1.49



Heavy duty grounded power cord and mating chassis connectors.

SOLAR CELLS



$1.19

10	100
$.98	$.85

125mA. at .43V. Ideal for powering radio receivers or charging batteries.

$.98



500mA "AA" FAST CHARGE

DISCOUNT Wire Wrap Center

IC SOCKETS

pin	Wire Wrap ea.	25	50	Solder ea.	25	50
8	—	—	—	17¢	16	15
14	37¢	36	35	20	19	18
16	38	37	36	21	20	19
24	99	93	85	36	35	34
40	1.69	1.55	1.39	63	60	58

KYNAR WIRE WRAP

50ft. **$.98**

500	1,000	11,000
$9.	$15.	$105.



$29.95

OK HOBBY WRAP-30 wire wrap & strip tool

$5.45

(213) 772-0800



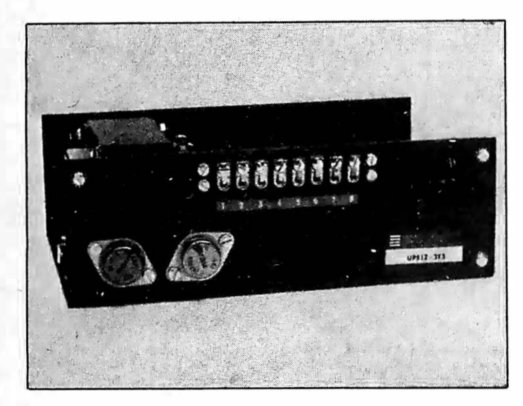


All merchandise sold by California Industrial is premium grade.
Orders are shipped the same day received.
PLEASE INCLUDE $1.00 SHIPPING ON ORDERS UNDER $15.00
California residents add 6% sales tax • Money back guarantee.
Sorry, no COD's • Foreign orders add 10%
Credit cards accepted, $20 minimum

Circle 21 on inquiry card.

What's New?

New Additions to UPS Uninterruptible Power Supply Family



Semiconductor Circuits Inc, 306 River St, Haverhill MA 01830, has announced the addition of 27 new models to its line of uninterruptible power supply systems. The UPS family float charges either a 12 or 24 V backup battery and offers the choice of a ±12 or ±15 VDC output for powering analog circuits. The third output, +5 VDC for powering logic, remains unchanged. MTBF (mean time between failures) is said to be in excess of 50,000 hours at 25°C.

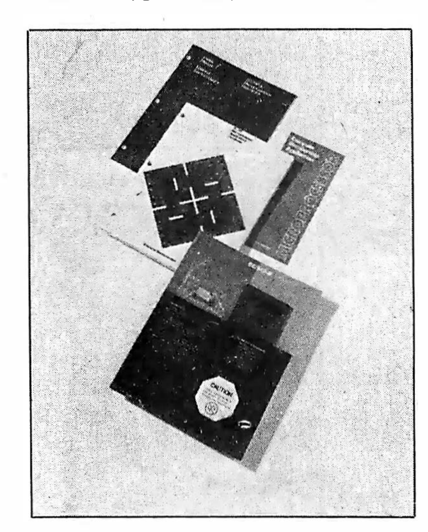
The UPS series conserves battery charge by employing both a DC/DC converter, which delivers either analog power outputs of ±12 or ±15 VDC at ±100, ±200 or ±300 mA, and a switching regulator, which delivers a logic power output of +5 VDC at 1, 2 or 3 A.

Under normal line conditions, series pass power supply serves as the power input to the DC/DC converters and as a float charge output for either 12 or 24 V backup batteries.

All models are packaged in a black anodized aluminum U-type open frame that is drilled for mounting and which measures 3 by 6 by 9 inches (7.62 by 15.24 by 22.86 cm). Prices range from $155 to $225 for a single unit, and $148 to $205 in quantities of ten and more. Availability is stock to two weeks.■

Circle 470 on inquiry card.

National Upgrades SC/MP Electronics



Faster, lower power n channel metal oxide semiconductor versions of National Semiconductor Corp's SC/MP microprocessor are now available as retrofits for SC/MP kits.

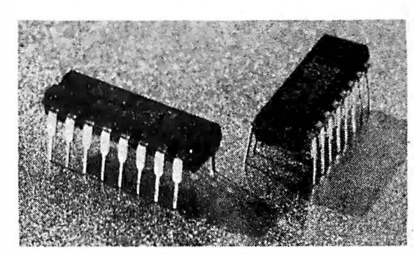
Called the SC/MP-II, the new 8 bit single chip device has all the features of the original p channel MOS version but will operate at twice the speed and will dissipate less than 200 milliwatts of power, about 25% of the power dissipated by the first SC/MPs introduced last year.

SC/MP-II requires only a +5 V supply, compared with the +5 and −7 V supplies required on earlier versions. Because of the +5 V only operation, the SC/MP-II can be interfaced with TTL and NMOS devices, and (by using pull up resistors) with CMOS devices.

The SC/MP-II microprocessor retrofit kit is available for $18.50. It includes the new SC/MP-II central processing unit (CPU), a 2 MHz crystal, a retrofit kit user's manual, an applications handbook and a SC/MP-II data sheet. No software changes are required as long as the retrofitted SC/MP-II runs at the same speed as its predecessor. Contact National Semiconductor at 2900 Semiconductor Dr, Santa Clara CA 95051.■

Circle 471 on inquiry card.

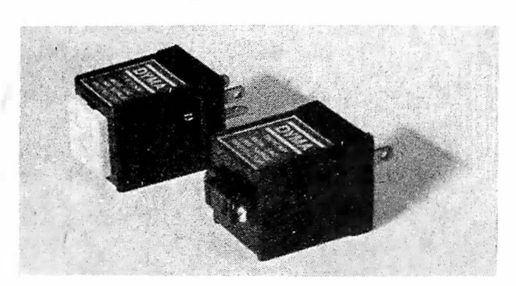
A Single Chip Stepper Motor Drive



North American Philips Controls Corporation, Cheshire Industrial Park, Cheshire CT 06410, (203) 272-0301, has introduced this integrated circuit stepper motor driver in a 16 pin dual in line package. The chip is intended to be used with 4 phase stepper motors which use 12 VDC and have 350 mA coils for each phase. This drive circuit includes the necessary logic to create motor motion in forward or reverse direction at rates determined by a clock input. The motors which North American Philips manufactures are listed in the brochure describing this part, and can typically provide working torque values in the .16 oz-in to 6 oz-in range with maximum stepping rates from 700 steps per second (lower torque motors) to about 200 steps per second (higher torque motors.) Typical step sizes for the motors mentioned in the engineering notes on the driver are 7.5° and 15°. With gearing, this type of motor should prove quite useful for robotic mechanisms experiments. Price for the SAA1027 driver circuit is $4.75 in lots of 100.■

Circle 473 on inquiry card.

No Transients Allowed



The Dyma AC Line Surge Protector is a suppressor and filter combination which is designed to protect equipment such as microprocessors and peripheral units from voltage transients on incoming power lines. The unit plugs directly into any standard AC outlet; equipment to be protected is plugged directly into the surge protector.

The 20 A load model is priced at $14.95. Other ratings are available on special order. Contact Dyma Engineering, 213 Pueblo Del Sur, POB 1697, Taos NM 87571.■

Circle 472 on inquiry card.

A New Music System Program

Software Technology Corporation has announced the Music System, a hardware and software package designed to generate music by producing three simultaneous tones of fixed amplitude using a complex waveform which approximates the sound of a reed organ. Tones are generated using square waves, which are actually produced by a highly controlled pulsing of one of the Altair (S-100) bus status lines.

The Music System comes complete with a program on cassette tape, six sample selections, a user's manual and a circuit board with components.

Running in close to 2 K bytes of programmable memory, the program includes a monitor, text editor compatible with Processor Technology's ALS-8 file structure, and a high level music composing language compiler. Language capabilites include dotted notes, 4 octave range and staccato.

With the addition of amplifier, speaker, cable and any Altair (S-100) bus computer, the Music System is ready to play. The price is $24.50. Contact Software Technology Corporation, POB 5260, San Mateo CA 94402, (415) 349-8080.■

Circle 474 on inquiry card.

What's New?

Game Theory

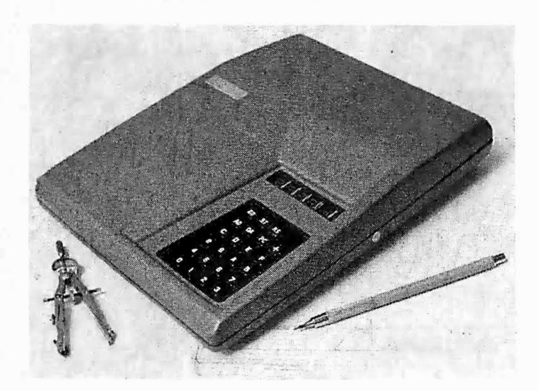


Tired of Monopoly, Aggravation and Sorry? Looking for a game that teaches something about computers as well as being fun? Then try Computer Rage for a change. First of all it uses three dice, but they're *binary* dice, so you can move from zero to seven spaces per turn. There are priority interrupts, input and output channels with finite capacity, power failures, program bugs and branch points. Your objective is to get your three programs (shaped like miniature disk packs) from the input to the output weaving through a maze of program steps, checkpoints, IO queues, interrupts and decision points.

Computer Rage comes with a large (19 by 19 inch) game board, 12 playing pieces, three binary dice, 38 interrupt cards, rules and a booklet describing how to use the game as an educational tool. Recommended for ages 9 to adult, two to four players. Several playing variations are possible. Computer Rage is available for $8.95 postpaid from Creative Computing, attn: Pamela, POB 789-M, Morristown NJ 07960.■

Circle 490 on inquiry card.

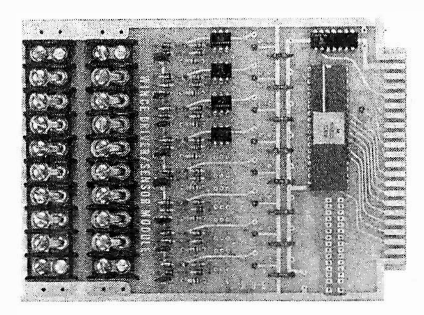
A Slick Dress for KIM-1

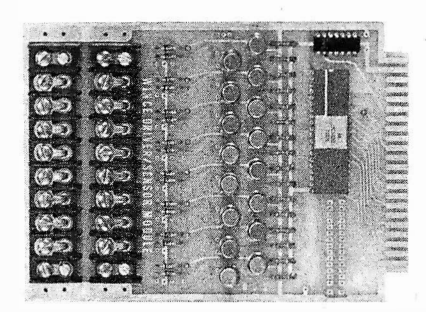


The Enclosures Group, 55 Stevenson St, San Francisco CA 94105, (415) 495-6925, has introduced this interesting enclosure for the KIM-1 product of MOS Technology. It should help to protect the circuit board of the KIM, especially during transit. The SKE 1-1 is available from stock in a variety of colors for $23.50.■

Circle 491 on inquiry card.

So You Want to Automate Your House?

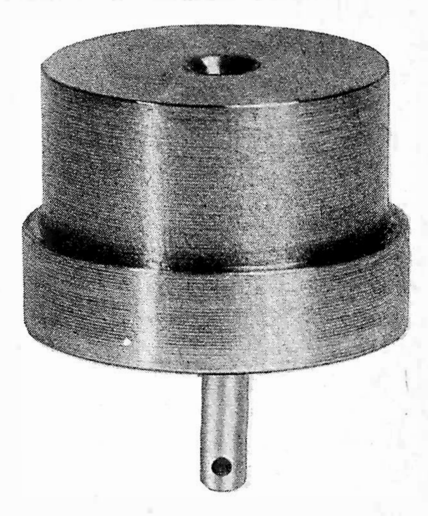




If you want to control things with a microprocessor system, boards like this product from Wintek, 902 N 9th St, Lafayette IN 47904, (317) 742-6802, will prove useful when applied with other products in the firm's line of modules. This photo shows the same board populated in two different ways to emphasize the fact that combinations of up to 16 output driver circuits or eight sensor inputs can be built on the same board, when ordered at a price of $69 plus $3 per driver and $12 per sensor. Drivers will handle up to 28 volts at 250 mA for use with relays, and sensors are optically isolated inputs for AC or DC voltages up to 240 V.■

Circle 492 on inquiry card.

Robot and Mechanism Hackers



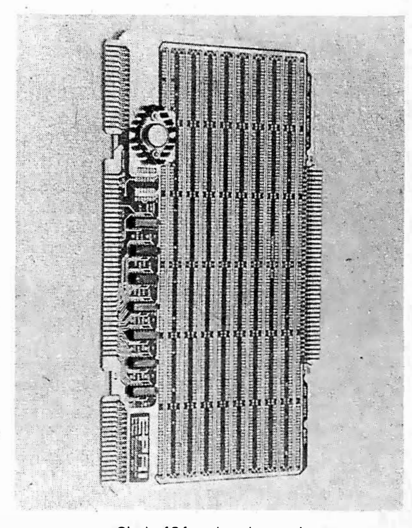
Here's an unusual item: Artisan Electronics has announced a new miniature solenoid designed with body dimensions equivalent to that of the T0-5 transistor case. Most applications for this T0-5 are for impulse duty, ie: the generation of relatively high forces for short times or pulsed operations on intermittent duty. On such impulse duty, the average power should not exceed ¾ W. Instantaneous power may be as high as 200 W, provided that the on time does not exceed 25 ms. At this duty, forces up to 50 grams may be generated at gaps of 0.100 inches (0.254 cm). For applications of continuous duty, the T0-5 solenoid will develop forces of from 1 to 10 grams with plunger travels up to .050 inches (0.025 cm). At this duty the solenoid is rated at ¾ W. A typical coil for operation on 12 VDC impulses would have a resistance of 1.5 Ω, pulsed at 12 VDC with a maximum on time of 25 ms and a minimum off time equal to 130 times the on time.

Contact Alan Seman, Artisan Electronics, 5 Eastmans Rd, Parsippany NJ, 07054, (201) 575-7684.■

Circle 493 on inquiry card.

3000 Hole General Purpose Prototyping Board

Electronic Product Associates Inc, 1157 Vega St, San Diego CA 92110, (714) 276-8911, announces the availability of a new general purpose prototyping board for use in the Micro-68 microprocessor systems. The 8 by 14.8 inch (20.3 by 37.6 cm) GP-2 board is Motorola Exorcisor bus compatible and has complete bus buffering already established using 8833 driver/receiver integrated circuits. The GP-2 board contains +5 V power and ground busing, 3000 holes worth of blank DIP patterns which allow for up to 35 large (24, 40 or 42 pin) DIP packages, or up to 107 small (14 or 16 pin) DIP packages. Price is $170, and they are said to be available from stock.■



Circle 494 on inquiry card.

D.R.C. ELECTRONICS

16K E-PROM CARD

S-100 (IMSAI/ALTAIR) BUSS COMPATIBLE

$69.95 (KIT)

IMAGINE HAVING 16K OF SOFTWARE ON LINE AT ALL TIME!

KIT FEATURES:

1. Double sided PC Board with solder mask and silk screen and Gold plated contact fingers.
2. Selectable wait states.
3. **All** address lines and data lines buffered!
4. All sockets included.
5. On card regulators.

USES 2708's!

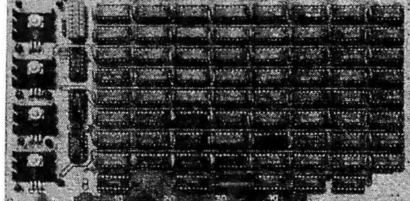
KIT INCLUDES ALL PARTS AND SOCKETS! (EXCEPT 2708's)

ADD $25 FOR ASSEMBLED AND TESTED

DEALER INQUIRES INVITED

SPECIAL OFFER: Our 2708's (650 NS) are **$12.95** when purchased with above kit.

8K LOW POWER RAM KIT!



$149.00 KIT

ADD $30 FOR ASSEMBLED AND TESTED. KIT FEATURES:

1. Double sided PC Board with solder mask and silk screen layout. Gold plated contact fingers.
2. All sockets included!
3. **Fully** buffered on all address and data lines.
4. **Phantom** is jumper selectable to pin 67.
5. **FOUR** 7805 regulators are provided on card.

S-100 (IMSAI/ALTAIR) BUSS COMPATIBLE

USES 21LO2-1 RAM'S.

S-100 REVERSING EXTENDER BOARD

$24.95 with connector

Turns the board under test around so that the foil side is facing you. Makes trouble shooting and debugging a SNAP! P.C. Layout designed to minimize noise and stray capacitance.

COMPUTER GRADE CAP.
48,000 MFD 25 WVDC Mallory
$3.95 **NEW!**

IC SOCKETS
For the newer RAM chips.
18 PIN — **4 FOR $1**
22 PIN — **3 FOR $1**

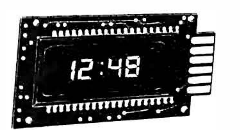
RCA HOUSE #2N3772
NPN Power Transistor. 30 AMP. 150 W. VCEO-60. TO-3. Vastly out performs 2N3055. Reg. List $3.04
2 FOR $1

T. I. ASCII CHARACTER GENERATOR
TMS 4103 JC. 28 PIN CER DIP. Has seven bit COLUMN Output for use with Matrix hard copy devices. With specs.
$3.50

NEW! 4K STATIC RAM'S NEW!
2114. The industry standard. 18 PIN DIP. Arranged as 1K X 4. Equivalent to FOUR 21LO2's in ONE package! TWO chips give 1K X 8, with data.
2 FOR $24
450 N.S.!

MOTOROLA 7805R VOLTAGE REGULATOR
Same as standard 7805 except 750 MA OUTPUT. TO-220. 5VDC OUTPUT.
$ .44 each **10 FOR $3.95**

NATIONAL SEMI. MA1003 CAR CLOCK
Not a kit. Complete tested module. Works on 12 VDC, has on board time base. Sold by others at $24.95. Big .30" Bright Green Digits. Same as used by Detroit in new cards.



$19.95

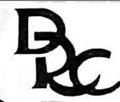
EDGE CONNECTOR — $1.50

Z - 80 PROGRAMMING MANUAL
By MOSTEK, the major Z - 80 second source. The most detailed explanation ever on the workings of the Z - 80 CPU CHIPS. At least one full page on **each** of the 158 Z - 80 instructions. A **MUST** reference manual for any user of the Z - 80. 300 pages. Just off the press! A D.R.C. exclusive! **$12.95**

2708 1KX8 EPROMS 2708
Prime new units from a major U.S. mfg. 650 N.S. access time. Equivalent to four 1702A's in one package!
$15.75 each

TERMS: ORDERS UNDER $15 ADD $ .75. NO C.O.D. WE ACCEPT VISA, MASTER CHARGE AND AMERICAN EXPRESS CARDS. MONEY BACK GUARANTEE ON ALL ITEMS. TEXAS RESIDENTS ADD 5% SALES TAX.

WE PAY POSTAGE!

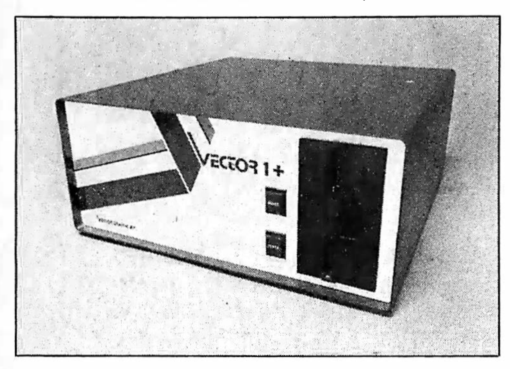


ELECTRONICS P. O. BOX 401247 • GARLAND, TEXAS 75040 • (214) 271-2461

What's New?

SYSTEMS

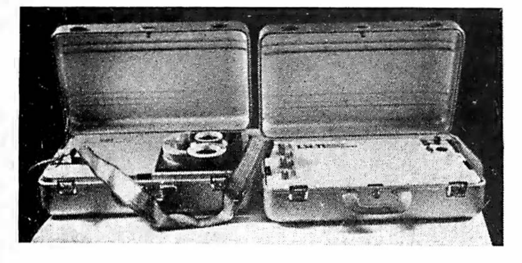
A Finished Product Concept



Vector Graphic Inc, 790 Hampshire Rd A-B, Westlake Village CA 91361, (805) 497-0733, has sent along this photo of the latest output of its design and production facility. This Vector 1+ is an Altair (S-100) bus computer product in an attractive cabinet, with provisions for the user to add a Shugart minifloppy™ (or equivalent) disk drive (not included), thus providing an integrated processor and mass storage combination rarely seen so far in the personal computing marketplace. Prices start at $659.■

Circle 475 on inquiry card.

An LSI-11 Based Computer in a Suitcase



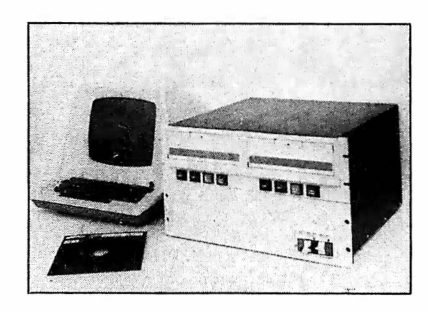
From RDA Inc comes news of the PRD11, an LSI-11 based microcomputer with the capacity for 56 K bytes of programmable memory and provisions for multiple terminal interfaces, a mass memory interface and a data acquisition subsystem. The entire unit weighs 23 pounds (10.43 kg) and is housed in an aluminum suitcase, a useful feature for the traveller.

Pictured with the PRD11 is a Computer Operations portable LINC tape mass memory compatible with the Digital Equipment Corporation's RT11 operating system. Software available includes a macroassembler, FORTRAN IV, multiuser BASIC, FOCAL and APL.

The PRD11 complete with 32 K bytes of programmable memory and a serial line interface is priced at $4,950. Contact RDA Inc, 5012 Herzel Pl, Beltsville MD 20705, (301) 937-2215.■

Circle 476 on inquiry card.

Andromeda's New Computer



Andromeda Systems has announced the Model 11/B, an LSI-11 based turnkey computer. The dual floppy system features 20 K by 16 bits of programmable memory and a 24 line video terminal with 80 characters per line. The terminal communicates with the computer via an RS232 interface at 9600 bps.

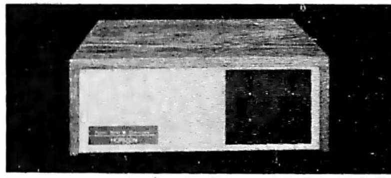
The 11/B uses the RT-11 operating system, the same system used by the Digital Equipment Corporation PDP-11 computer. It is designed for the single interactive user, although it can support up to eight users under multiuser BASIC (optional). System programs include a text editor, macroassembler, file manager and batch monitor. The user can choose from a variety of high level language options, including FORTRAN and FOCAL. The processor has a built-in floating point package.

The floppy disk system provides 512 K bytes of on line mass storage. A bootstrap loader program is built into the disk controller.

Contact Andromeda Systems, 14701 Arminta St #J, Panorama City CA 91402, (213) 781-6000.■

Circle 477 on inquiry card.

North Star's New Computer



North Star Computers Inc has announced the new North Star Horizon™ computer, which uses a full speed (4 MHz) Z-80 microprocessor and includes 16 K bytes of memory, a disk controller with one or two Shugart minifloppy™ disk drives, and full extended disk BASIC. A serial IO port is also provided.

Options include additional disk drives, hardware floating point arithmetic board, 24 line by 80 character upper and lower case video display controller board, and 16 K memory board with parity check. The video display board, when used in conjunction with the 16 K memory board, will display high resolution (480 by 250 point) graphics on a video monitor. The Horizon computer uses the Altair (S-100) bus.

The single drive is $1599 in kit form and $1899 assembled. The dual drive is $1999 in kit form and $2349 assembled. Contact North Star Computers Inc, 2465 Fourth St, Berkeley CA 94710, (415) 549-0858.■

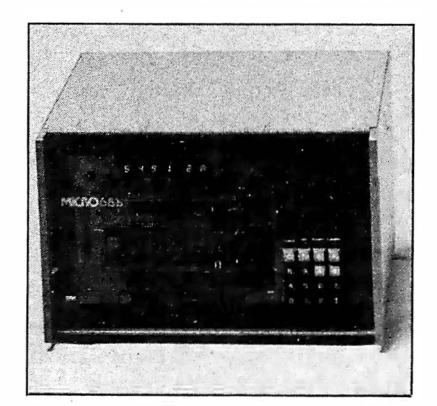
Circle 478 on inquiry card.

Attention London Computer Hackers

London's Computer Workshop has announced a new 4 terminal multiuser computer system including a printer and a BASIC compiler for under £3000. To obtain more information about this system, contact either Gordon Ashbee or John Burnett at the Computer Workshop, 174 Ifield Rd, London SW10 9AG ENGLAND, phone 01 373 8571.■

Circle 479 on inquiry card.

EPA's Microcomputer and Floppy Disk



Electronic Product Associates Inc, 1157 Vega St, San Diego CA 92110, have announced a combined microcomputer and floppy disk system which uses the 6800 processor. The microcomputer is designated the Micro-68b and comes complete with 8 K bytes of programmable memory plus the Motorola MIKBUG monitor system, 20 mA current loop and RS-232 interfaces, and cassette interface. In addition, there is a built-in hexadecimal keyboard and LED display.

The Micro-68 floppy disk system is compatible with IBM standards and is available in either single or dual configurations. Both versions come complete with power supply and interface electronics.

The Micro-68b costs $1878; the single floppy disk system is $2595, and the dual version is $3295. Software available includes FORTRAN IV, BASIC, assembler language, an editor, and a floppy disk operating system.■

Circle 480 on inquiry card.

APPLE II I/O BOARD KIT

Plugs Into Slot of Apple II Mother Board

FEATURES:

- 18 Bit Parallel Output Port (Expandable to 3 Ports)
- 1 Input Port
- 15mA Output Current Sink or Source
- TTL or CMOS Compatible
- Addressable anywhere in memory output area
- Can be used for peripheral equipment such as printers, floppy discs, cassettes, paper tapes, etc.

KIT INCLUDES:
P.C. Board, I.C.'s Sockets and Assembly Manual.

PRICE:
1 Input and 1 Output Port for $49.00

1 Input and 3 Output Ports for $64.00

DEALER INQUIRIES INVITED

UNGAR SOLDERING IRONS

27W SOLDERING IRON KIT	I.C. DESOLDERING KIT	10W ASSEMBLED SOLDERING IRON
Includes iron, 2 tips, roll of solder and iron stand	Includes iron and 3 desolder tips for dual in lines, cams, etc.	
$6.72	$24.70	$14.97



LD-130
3 DIGITS A/D CONVERTER
$11.95

PUSH BUTTON SWITCH
Red or Green

3 for $1.00

3RD GENERATION ASCII KEYBOARD KIT
ONLY $63.00

FURTHER IMPROVEMENTS, MORE FEATURES

- TTL Logic Circuits
- Power: +5V, 275mA
- Upper and Lower Case
- Full ASCII Set (Alpha Numeric, Symbols, Control)
- 7 ot 8 Bits Parallel Data
- Optional Serial Output
- Selectable Positive or Negative Strobe, and Strobe Pulse Width
- 'N' Key Roll-Over
- Full Debounced
- Carriage Return Key
- Repeat Function Key
- Shift Lock, 2 Shift Keys
- 4 User Defineable Keys
- P.C. Board Size: 17-3/16" x 5"

OPTIONS:

- Metal Enclosure (Painted IBM Blue and White $25.00
- 18 Pin Edge Con. $2.00
- I.C. Sockets . . . $4.00
- Serial Output (Shift Register) $2.00
- Upper Case Lock Switch (for Capital Letters and Numbers) $2.00

KIT NUMBERS: Keyboard, P.C. Board, all required components and assembly manual.

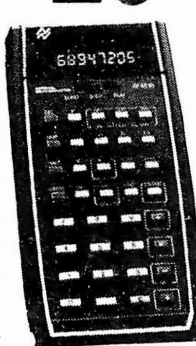
NOTE: If you have this 63 Key Teletype Keyboard you can buy the Kit without it for $44.95.

A FULLY PROGRAMMABLE SLIDE RULE

the MATHEMATICIAN with 100 STEPS

$29.95





- RPN logic with "built-in" hierarchy for increased speed and accuracy in calculating sequences involving arithmetic, trigonometric, logarithmic, power or exponential functions.
- A three-level stack plus separate accumulating memory for quick, accurate solutions to complex calculations.
- Eight-digit LED display with full-floating decimal system.
- Common and natural logarithms and antilogarithms.
- Sine, cosine, tangent and inverse trigonometric functions.
- Instant automatic calculation of powers and roots.
- Instant conversions of radians to degrees or vice versa.
- Square, square root, and reciprocal calculations.
- Pi, change sign, and register exchange keys.
- Automatic reciprocals.
- Ability to automatically sum squares.
- Storage memory.
- "Roll-down" clear.
- MOS/LSI solid-state circuitry.
- Engineered and manufactured by National Semiconductor Corp., a world leader in solid-state technology.
- Simplified programming. You simply engage a learn switch and perform a problem in normal manner. The 4615 records the formula and lets you debug the program as it's written.
- The learn-mode capacity total 100 separate steps.
- Several different programs can be contained at the same time.
- Constant factors can be entered as program steps.
- Delete feature lets you correct programs while you are writing them.
- Skip key permits skipping over entire programs to access additional programs within 100-step capacity.
- Programs remain intact until new programs are written over or until your 4615 is turned off.
- You have total freedom to select keyboard entries as variables or constants.
- Automatic warning signal in display lets you know when you exceed programming capacity.
- The 4615 is rechargeable and comes complete with nickel cadmium batteries.

10 DAY MONEY BACK GUARANTEE

___ 4615 @ 29.95 ______

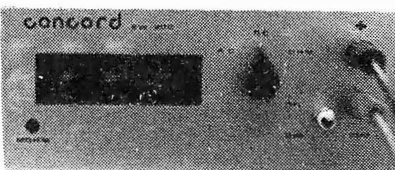
___ AC charger @ 4.95 ______

___ Carrying case @ 2.95 ______

___ Calc. stand @ 1.95 ______

FROM CONCORD

THE FIRST FULL *FEATURE* LSI DMM KIT



INTRODUCTORY PRICE: $77.77

Reg. Suggested Retail: $149.00

- AUTO RANGING
- AUTO POLARITY
- AUTO ZERO
- 3 Large Digits (1/2")
- Rechargable

MEASUREMENT RANGES:
Voltage: (AC & DC) 1 MV - 1000V
Current: (AC & DC) 10 μA - 1A
RESISTANCE: 1 Ω - 10 M Ω
Basic D.C. Accuracy, better than 0.1% ± 1 Digit
Power: 4 AA batteries (Rechargable batteries optional)

NI-CAD BATTERIES: $6.00 • AC CHARGER: $4.95 • ENCLOSURE: $12.95 • TEST LEADS: $2.95 • SHUNT KIT FOR 3 CURRENT RANGES: $4.75

Minimum Order: $10.00

SHIPPING AND HANDLING
KEYBOARD, DMM and CALCULATOR: $3.50.
ALL OTHERS – $1.00.

California residents add 6% sales tax

ELECTRONICS WAREHOUSE Inc.

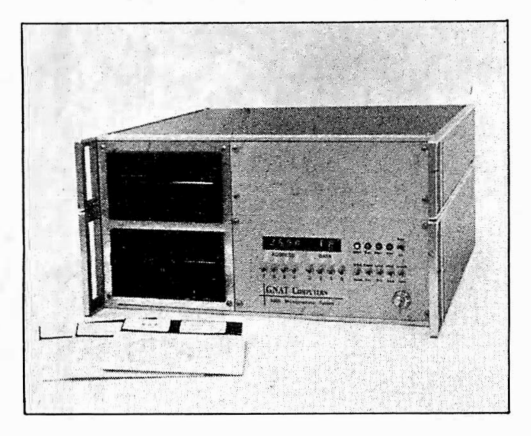
**1603 AVIATION BLVD.
REDONDO BEACH, CA. 90278
TEL. (213) 376-8005
WRITE FOR FREE CATALOG**

You are invited to visit our store at the above address

What's New?

SYSTEMS

Gnat Leaps into Dual Minifloppy System



The dual GNAT-PAC System 8 microcomputer is now available with dual minifloppy disk drive from Gnat Computers Inc, 7895 Convoy Ct, Unit 6, San Diego CA 92111. Each minifloppy has storage capacity of 80 K bytes; operations include on board data buffering, automatic seeking and disk initialization.

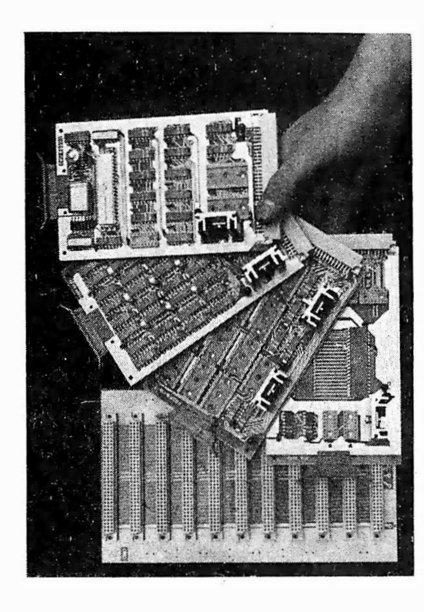
The system is called the extended System 8, and it comes equipped with the following features:

- 16 K bytes of programmable memory.
- 2 K bytes of programmable read only memory (PROM) with space for an additional 14 K bytes.
- Serial and parallel IO.
- Disk interface and controller.
- Hexadecimal front panel.

Software for the System 8 includes a monitor, bootstrap loader, and disk operating system. The monitor and loader are PROM resident. The disk operating system features an assembler, editor and debugger with trace, test and debug capability. PL/M, BASIC, FORTRAN and other high level languages are available. Single unit price is $3690.■

Circle 462 on inquiry card.

A "Puzzling" New Development from Europe



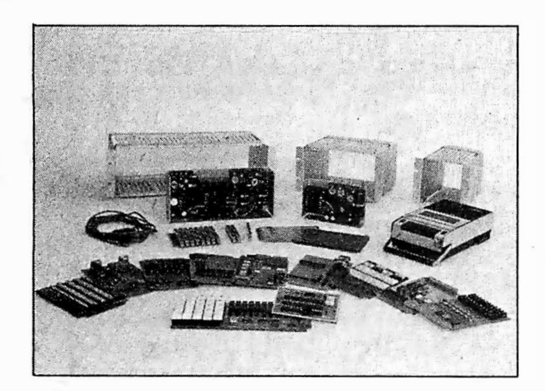
What's happening in Europe? One answer to the question is ES's new "Puzzle" microprocessor system. Puzzle is a 6502 based system which uses European style printed circuit boards and connectors. It consists of a processor card (with 500 bytes of programmable memory and 2 K bytes of programmable read only memory), a 4 K byte programmable memory extension card, a 6 to 12 K byte programmable memory extension card, and an IO card.

A software development and test system is also available which features serial and parallel interfaces.

For more information, contact Ing Ernst Steiner, 1130 Wein Geylinggasse 16 AUSTRIA, phone 82 26 74.■

Circle 463 on inquiry card.

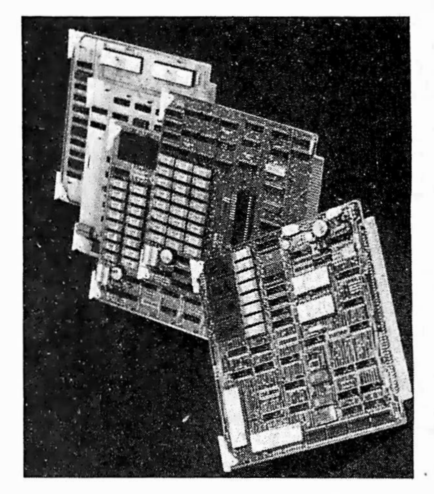
Wintek Module Line



Here is the beauty contest photo of the Wintek line of modules for microprocessor systems use. This unique line of "Wince Micro Modules" includes backplanes, card racks, power supplies and associated items. The intent is to allow the user to quickly and cheaply assemble a customized microcomputer system using the standard 4.5 by 6.5 inch (11.4 by 16.5 cm) 44 pin connector cards available from the firm. Options available include modules for control, volatile user memory, ROM and EROM programmer, analog interfaces, data acquisition, relay drivers and sense inputs, cassette interfaces, floppy disk interface, console, CMOS volatile memory with battery backup, and a touch tone transmit/receive modem. Wintek is located at 902 N 9th St, Lafayette IN 47904, (317) 742-6802.■

Circle 464 on inquiry card.

Z80 Microcomputer Boards



Zilog Inc has introduced a family of Z80 based microcomputer boards to offer users a modular approach for building their own computing and processing systems.

The new MCB board series is designed with Zilog's Z80 circuit. Each of the present boards is bus compatible and directly interfaces with all other boards in the series. All boards currently offered are available for delivery 30 days ARO.

Leading off the series is the Z80-MCB microcomputer board, designed to operate as a single board computer, including its own self-contained memory, plus serial and parallel IO ports. The Z80-MCB has capacity of 4 K bytes of dynamic programmable memory, plus up to 4K bytes of programmable read only memory, erasable read only memory, or read only memory.

The Z80-MCB can be expanded to include more IO and memory by adding other boards in the series. A strapping option allows users to put 16 K by 1 bit dynamic programmable memory modules in place of 4 K by 1 bit programmable memory modules. Single unit price of the Z80-MCB is $495.

The Z80-MDC memory/disk controller board provides users with 12 K bytes of dynamic programmable memory, plus a floppy disk controller capable of handling up to eight floppy disk drives. The Z80-MDC has a strapping option for setting start address of each 4 K byte page. Another feature is a 16 bit cyclic redundancy check. Single quantity price of the Z80-MDC is $795.

The Z80-PMB/PROM memory board provides up to 32 K bytes of memory. Jumper options allow each 16 K bytes of memory to reside in any segment of the 64 K address space. The price is $395 for single units, excluding the cost of programmable read only memory chips.

Also included in the MCB series are three programmer boards and three interface boards. Standard card cages, extender boards, edge connectors and wire wrap boards are available as options.

For more information contact Dave West at Zilog, 10460 Bubb Rd, Cupertino CA 95014, (408) 446-4666.■

Circle 465 on inquiry card.

From Parts to Peripherals:
a one stop, mail-order computer store, serving computer enthusiasts since 1973.

Vector VP2 Assembled Microcomputer Case

This adjustable packaging system for S-100 buss microcomputers is compatible with Altair 8800 and IMSAI 8080 size cards. Outside, it is beautiful...with a dark blue textured vinyl finish and lines unmarred by external screws or fasteners. Inside, there is space for 21 cards total (on 0.75" centers) with a fully adjustable interior card mounting system (card guides and hardware provided for 12 cards). The interior is instantly accessible, the rear and front panels are removable and recessed. If you want a classy home for your micro, check this out...it is the best we have seen.

VP2 ASSEMBLED MICROCOMPUTER CASE....$134.30

CRYSTALS

STATEK 3 TERMINAL CRYSTALS.......$4.95 each
(all frequencies in KHz)

10.000	12.800	15.360	16.000	16.384
18.641	19.200	20.480	24.576	30.720
31.500	32.768	36.864	38.400	40.960
60.000	76.800	100.00	153.60	240.00

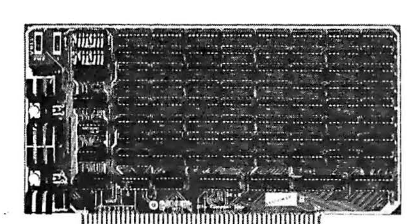
SENTRY CRYSTALS: SERIES MODE, FUNDAMENTAL, WIRE LEADS, HC18 PACKAGE.........$4.95 each
(all frequencies in MHz)

4.0	4.5315	5.0	8.0	9.0
10.0	12.0	15.0	18.0	20.0

MISCELLANEOUS CRYSTALS

500 KHz, series mode, fundamental, HC6/U package, wire leads.............**$4.95 each**
1 MHz, series mode, fundamental, wire leads in HC6/U package.................**$5.95 each**
2 MHz, series mode, fundamental, wire leads in HC6/U package.................**$5.90 each**

8Kx8 Econoram II™
single kit $163.84
3 kits-24K!-$450



Those who know memory recognize the Godbout board as not just an exceptional value (it's no secret we know how to keep costs down), but as an example of how to pack extra options into a basic memory board. Extras like a vector interrupt provision if you try to write into protected memory. Configuration as **two** independent 4K blocks (both protectable separately). A selectable write strobe for either PWR or MWRITE. An all static design, free of timing and refresh problems. Guaranteed speed under 450 ns (with on-board wait state logic for use with 4 MHz Z-80) and guaranteed current under 1.5A (1250 mA typ). And of course...sockets for all ICs, legended board with solder mask, one year warranty on parts...**we've got it all.**

ALSO AVAILABLE: 8K ASSEMBLED, TESTED, WARRANTED 1 YEAR...$188.50; 4K KIT...$100.

Low power Schottky

74LS00	$0.30	74LS151	$0.95
74LS01	0.30	74LS155	1.38
74LS02	0.30	74LS157	0.95
74LS04	0.33	74LS160	1.40
74LS08	0.36	74LS161	1.40
74LS10	0.30	74LS162	1.40
74LS11	0.36	74LS163	1.40
74LS12	0.33	74LS168	1.87
74LS14	1.38	74LS169	1.87
74LS15	0.30	74LS173	1.65
74LS20	0.30	74LS174	1.25
74LS21	0.33	74LS175	1.15
74LS22	0.33	74LS195	1.30
74LS26	0.43	74LS240	1.88
74LS27	0.36	74LS257	1.25
74LS30	0.30	74LS258	1.25
74LS32	0.38	74LS266	0.53
74LS37	0.45	74LS283	1.20
74LS38	0.45	74LS365/	
74LS42	0.98	80LS95	0.75
74LS47	1.00	74LS366/	
74LS48	0.98	80LS96	0.75
74LS74	0.50	74LS367/	
74LS75	0.68	80LS97	0.75
74LS76	0.50	74LS368/	
74LS86	0.50	80LS98	0.75
74LS109	0.50	74LS386	0.55
74LS125	0.63	81LS95	1.13
74LS126	0.63	81LS96	1.13
74LS132	1.25	81LS97	1.13
74LS138	1.10	81LS98	1.13
74LS139	1.15		

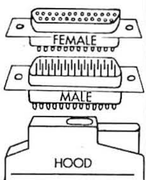


DB-25 CONNECTOR

25 PIN RS-232 CONNECTORS: submini D type.

Male plug with plastic hood, part #DB25P $3.95

Female jack, part #DB25S $3.95

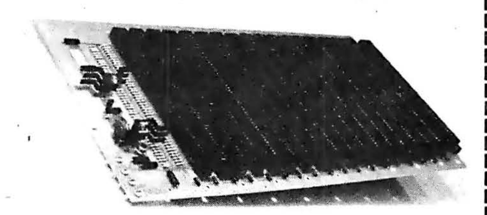


EDGE CONNECTORS: #S-100-140ST Gold plated soldertail edge connector, 0.140" spacing for Altair Motherboards....**1/$6, 5/$27.50**
#S-100ST same but .250" row spacing for IMSAI.....**1/$5, 5/$22.00**
#S-100WW same but gold plated, 3 level wrap posts..**1/$5, 5/$22.00**

Active Terminator $29.50

Plugs into any S-100 Motherboard whose buss lacks active terminations. Cleans up noise, crosstalk, overshoot, and other buss problems that can scramble data unpredictably. Kit form only.

MOTHERBOARD
10 Slots $90   18 Slots $124



Includes **all** edge connectors, plus active terminations to minimize crosstalk, noise, overshoot, and ringing that may be present with unterminated busses. Excellent for stand alone system, or add to existing systems. Kit form only.

Music spoken here

Many of you are into making music with your computer; some people generate sounds within the machine itself, others use them to control synthesizer hardware. Where do we come in? One of our lines is the series of MusiKits™ designed by Craig Anderton, writer for Guitar Player, Popular Electronics, Contemporary Keyboard, and a bunch of others. MusiKits™ contain a circuit board, electronic components, and pots; user supplies case and hardware. There are 24 in all...some you might like are the **8 In, 1 Out Mixer (#18, $20)** which can mix up to 8 audio inputs to a common output; the **Super Tone Control (#17, $10.50)**, a low noise, state variable filter that gives high, low, and bandpass outputs; a **Reverb Unit (#22, $13.00** less springs) for adding concert hall sound and acoustic depth. Want to know more? Check our flyer.

SPECIAL 2/$1.25

10000 uF at 10V!

Building a 5V power supply? Need to beef up an existing supply that lacks sufficient capacitance? Here is the answer.

TERMS: Please allow up to 5% for shipping, more for Vector #VP 2; excess refunded. Prices good through end of magazine cover month. Californians add tax. CODs accepted with street address. For BankAmericard®/VISA®/Mastercharge® orders ($15 min) call 415-562-0636, 24 hours.

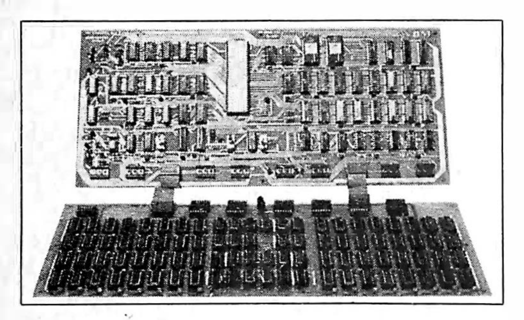
GODBOUT

BILL GODBOUT ELECTRONICS
BOX 2355, OAKLAND AIRPORT, CA 94614

SEVERAL GOOD REASONS WHY YOU SHOULD HAVE OUR FLYER: 1) CMOS 2) LINEARS 3) MICROPROCESSORS 4) POWER SUPPLIES 5) RESISTORS 6) CAPACITORS 7) DISPLAYS 8) SOCKETS 9) VECTOR PRODUCTS 10) ENCLOSURES 11) ALL THE OTHER THINGS WE CAN'T FIT INTO THIS SPACE.

Circle 61 on inquiry card.

Will Memory Megalomania Never Stop?



Technico Inc, 9130 Red Branch Rd, Columbia MD 21045, (800) 638-2893, has just sent along an exciting new development for those readers interested in opting for the Texas Instruments TMS-9900 architecture. The firm manufactures a TMS-9900 based processor board, shown at the top in this photo. The new development is the board at the bottom, the TEC-9900-MA dynamic memory board which has a capacity of up to 32 K bytes (16 K 16 bit words) and plugs directly into the previous TEC-9900-SS product. The new board measures 7 by 16 inches (17.8 by 40.6 cm) and includes all the necessary refresh and control circuitry. Address selection logic allows DIP switches to specify any starting address for the 32 K byte region in 1 K address increments. The board uses Texas Instruments TMS-4051 dynamic memory chips which are organized 4 K by 1 bit. As a result, the board can be populated with any increment of 4 K 16 bit words. These memory parts have a sufficiently fast response time to allow the 9900 to run at its maximum speed of 3.3 MHz clock. A fully populated 32 K byte version of the board sells under part number TEC9900-MA-32KB for $799 assembled and tested. (At the price of 3 cents/bit installed, who can haggle?) This is a very desirable item for the homebrewer willing to supply the finishing touches to a 16 bit minicomputer with 32 K bytes of memory. The economics can be summarized as follows:

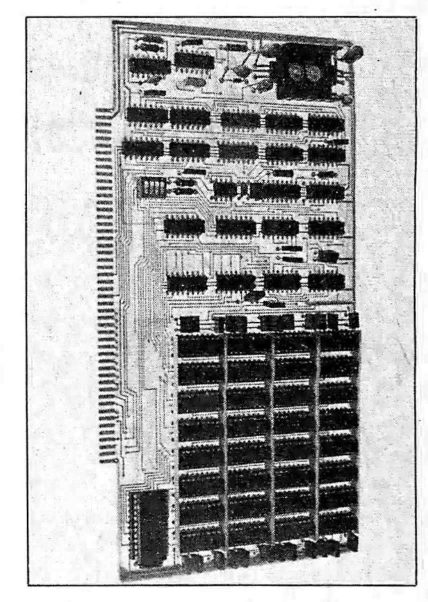
TEC-9900-SS Processor TI-9900 + ancilliaries	$399
TEC-9900-MA-32KB Memory 32,768 bytes memory	$799
TEC-9900-PP Power supply	$149
	$1347

To these assembled and tested module prices one should add the cost of a serial ASCII terminal, a cabinet or chassis in which to mount the equipment (homebrew style), and any mass storage required to complete the system.

A phone conversation with the firm at the time this note was being written (July 5 1977) brought out the fact that forthcoming additions to the line are a video and audio cassette interface board (TEC-9900-VA), and a floppy disk controller, both of which were expected to be available in the fourth quarter of 1977.■

Circle 466 on inquiry card.

Memory Module from MITS



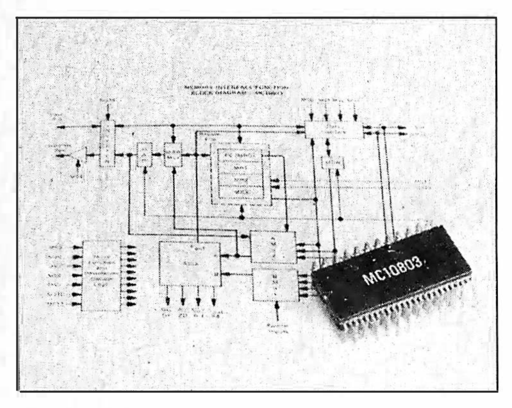
The new Altair memory module provides 16 K bytes of dynamic random access memory. The unit runs at a maximum power dissipation of 3 W and a maximum cycle time of 350 ns.

Crystal controlled logic timing eliminates the need for on board oneshot multivibrator circuitry to allow continuous operation without wait states.

Bus strips provide isolation between power and signal lines for maximum noise suppression. Address selection is switch selectable in 4 K blocks. Each board requires one slot on the Altair (S-100) 8800 bus. Contact MITS at 2450 Alamo SE, Albuquerque NM 87106.■

Circle 467 on inquiry card.

Attention Microprogrammed Computer Designers



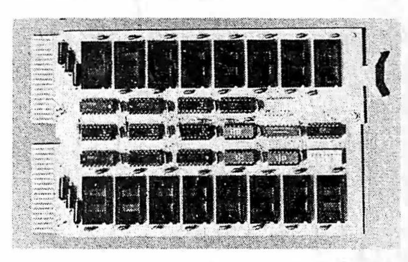
Motorola has sent along this photo and block diagram of a new addition to their M10800 family of high speed MECL 10,000 current mode logic. (This family of logic is used in the highest speed contemporary processors, and due to the difficulties of designing with transmission line interconnections, tends to be ignored by experimenters favoring slower TTL logic.) Illustrating the way future packaging trends are going, this MC10803 memory interface processor is mounted in a so-called "quad in line" (QUIL) package with four rows of 12 pins for 48 pins total.

Its internal logic, shown in the background, includes six 4 bit registers, an arithmetic logic unit (with encoded selections of function and operands) and data transfer circuitry. Its intended use is as a node in a large machine, dedicated to memory and peripheral operations. An example is performing the tasks of direct memory access control where intelligent programming is useful, but the versatility of a main processor is not needed.

The high speed nature of this device and its price ($40 in 100 quantity) say that this chip will be of most interest to those individuals designing new products subject to high performance specifications. For more information contact Jerry Tonn at Motorola Semiconductor Products, POB 20912, Phoenix AZ 85036, (602) 962-2515.■

Circle 468 on inquiry card.

An LSI-11 EROM Board



RDA Inc, 5012 Herzel Pl, Beltsville MD 20705, (301) 937-2215, has sent along this picture of an LSI-11 option (Digital Equipment Corporation) which requires two slots of an LSI-11's backplane and provides 8 K bytes of storage using 2708s as memory elements. The memory is set up for selectable addressing. Power requirements are 1 A on each of three voltages: +5 V, −5 V and −12 V when all sixteen 2708 memories are plugged into the circuit in the empty sockets shown in the photo. (A separate programming device is required since this board does not include a built-in programmer.) Exclusive of the 16 memory chips, the price of this board is $285. Assuming a current mail order price of $35 for each 2708, fully stuffing the board will cost $560. ■

Circle 469 on inquiry card.

DIODES/ZENERS

1N914	100v	10mA	.05
1N4005	600v	1A	.08
1N4007	1000v	1A	.15
1N4148	75v	10mA	.05
1N753A	6.2v	z	.25
1N758A	10v	z	.25
1N759A	12v	z	.25
1N4733	5.1v	z	.25
1N5243	13v	z	.25
1N5244B	14v	z	.25
1N5245B	15v	z	.25

SOCKETS/BRIDGES

8-pin	pcb	.25	ww	.45
14-pin	pcb	.25	ww	.40
16-pin	pcb	.25	ww	.40
18-pin	pcb	.25	ww	.75
22-pin	pcb	.45	ww	1.25
24-pin	pcb	.35	ww	1.10
28-pin	pcb	.35	ww	1.45
40-pin	pcb	.50	ww	1.25

Molex pins	.01	To-3 Sockets	.45
2 Amp Bridge		100-prv	1.20
25 Amp Bridge		200-prv	1.95

TRANSISTORS, LEDS, etc.

2N2222	NPN	(Plastic .10)	.15
2N2907	PNP		.15
2N3906	PNP		.10
2N3054	NPN		.35
2N3055	NPN	15A 60v	.50
T1P125	PNP	Darlington	.35
LED Green, Red, Clear			.15
D.L.747	7 seg 5/8" high com-anode		1.95
XAN72	7 seg com-anode		1.50
FND 359	Red 7 seg com-cathode		1.25

C MOS

4000	.15
4001	.20
4002	.20
4004	3.95
4006	1.20
4007	.35
4008	.95
4009	.30
4010	.45
4011	.20
4012	.20
4013	.40
4014	1.10
4015	.95
4016	.35
4017	1.10
4018	1.10
4019	.60
4020	.85
4021	1.35
4022	.95
4023	.25
4024	.75
4025	.35
4026	1.95
4027	.50
4028	.95
4030	.35
4033	1.50
4034	2.45
4035	1.25
4040	1.35
4041	.69
4042	.95
4043	.95
4044	.95
4046	1.75
4049	.70
4050	.50
4066	.95
4069	.40
4071	.35
4081	.70
4082	.45

— T T L —

7400	.15	7473	.25	74176	1.25	74H72	.55	74S133	.45
7401	.15	7474	.35	74180	.85	74H101	.75	74S140	.75
7402	.20	7475	.35	74181	2.25	74H103	.75	74S151	.35
7403	.20	7476	.30	74182	.95	74H106	.95	74S153	.35
7404	.15	7480	.55	74190	1.75			74S157	.80
7405	.25	7481	.75	74191	1.35	74L00	.35	74S158	.35
7406	.35	7483	.95	74192	1.65	74L02	.35	74S194	1.05
7407	.55	7485	.95	74193	.85	74L03	.30	74S257 (8123)	.25
7408	.25	7486	.30	74194	1.25	74L04	.35		
7409	.15	7489	1.35	74195	.95	74L10	.35	74LS00	.35
7410	.10	7490	.55	74196	1.25	74L20	.35	74LS01	.35
7411	.25	7491	.95	74197	1.25	74L30	.45	74LS02	.35
7412	.30	7492	.95	74198	2.35	74L47	1.95	74LS04	.35
7413	.45	7493	.40	74221	1.00	74L51	.45	74LS05	.45
7414	1.10	7494	1.25	74367	.85	74L55	.65	74LS08	.35
7416	.25	7495	.60			74L72	.45	74LS09	.35
7417	.40	7496	.80	75108A	.35	74L73	.40	74LS10	.35
7420	.15	74100	1.85	75110	.35	74L74	.45	74LS11	.35
7426	.30	74107	.35	75491	.50	74L75	.55	74LS20	.35
7427	.45	74121	.35	75492	.50	74L93	.55	74LS21	.25
7430	.15	74122	.55			74L123	.55	74LS22	.25
7432	.30	74123	.55	74H00	.25			74LS32	.40
7437	.35	74125	.45	74H01	.25	74S00	.55	74LS37	.35
7438	.35	74126	.35	74H04	.25	74S02	.55	74LS40	.45
7440	.25	74132	1.35	74H05	.25	74S03	.30	74LS42	1.10
7441	1.15	74141	1.00	74H08	.35	74S04	.35	74LS51	.50
7442	.45	74150	.85	74H10	.35	74S05	.35	74LS74	.65
7443	.85	74151	.75	74H11	.25	74S08	.35	74LS86	.65
7444	.45	74153	.95	74H15	.30	74S10	.35	74LS90	.95
7445	.65	74154	1.05	74H20	.30	74S11	.35	74LS93	.95
7446	.95	74156	.95	74H21	.25	74S20	.35	74LS107	.85
7447	.95	74157	.65	74H22	.40	74S40	.25	74LS123	1.00
7448	.70	74161	.85	74H30	.25	74S50	.25	74LS151	.95
7450	.25	74163	.95	74H40	.25	74S51	.45	74LS153	1.20
7451	.25	74164	.60	74H50	.25	74S64	.25	74LS157	.85
7453	.20	74165	1.50	74H51	.25	74S74	.40	74LS164	1.90
7454	.25	74166	1.35	74H52	.15	74S112	.90	74LS367	.85
7460	.40	74175	.80	74H53J	.25	74S114	1.30	74LS368	.85
7470	.45			74H55	.25				
7472	.40								

LINEARS, REGULATORS, etc.

8266	.35	LM320K5 (7905)	1.65	LM340T24	.95	LM723	.50
MCT2	.95	LM320K12	1.65	LM340K12	2.15	LM725	1.75
8038	3.95	LM320T5	1.65	LM340K15	1.25	LM739	1.50
LM201	.75	LM320T12	1.65	LM340K18	1.25	LM741 (8-14)	.25
LM301	.25	LM320T15	1.65	LM340K24	.95	LM747	1.10
LM308 (Mini)	.75	LM339	.95	LM373	2.95	LM1307	1.25
LM309H	.65	7805 (340T5)	.95	LM380	.95	LM1458	.95
LM309K (340K-5)	.85	LM340T12	1.00	LM709 (8,14 PIN)	.25	LM3900	.50
LM310	1.15	LM340T15	1.00	LM711	.45	LM75451	.65
LM311D (Mini)	.75	LM340T18	1.00			NE555	.50
LM318 (Mini)	.65					NE556	.95
						NE565	.95
						NE566	1.75
						NE567	1.35

9000 SERIES

9301	.85
9309	.35
9322	.85
95H03	.55
9601	.75
9602	.50

MEMORY CLOCKS

74S188 (8223)	3.00
1702A	6.95
MM5314	3.00
MM5316	3.50
2102-1	1.75
2102L-1	1.95
TR 1602B/ TMS 6011	6.95
8080AD	15.00
8T13	1.50
8T23	1.50
8T24	2.00
2107B-4	4.95

INTEGRATED CIRCUITS UNLIMITED

7889 Clairemont Mesa Boulevard, San Diego, California 92111
(714) 278-4394 *(Calif. Res.)*

All orders shipped prepaid — No minimum
Open accounts invited — COD orders accepted

Discounts available at OEM Quantities — California Residents add 6% Sales Tax
All IC's Prime/Guaranteed. All orders shipped same day received.

24 Hour Toll Free Phone 1-800-854-2211 — **MasterCharge / BankAmericard / AE**

SPECIAL DISCOUNTS

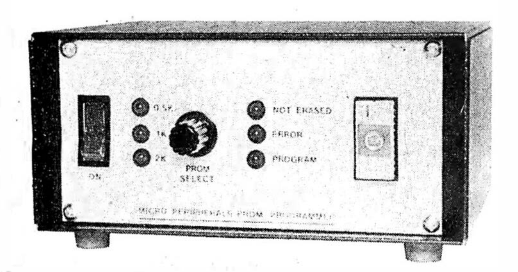
Total Order	Deduct
$35 - $99	5%
$100 - $300	10%
$301 - $1000	15%
$1000 - Up	20%

Circle 70 on inquiry card.

What's New?

MEMORY

An EROM Programmer



MicroPeripherals, 24 Matford Close, Westbury on Trym, Bristol BS10 6LR ENGLAND, has announced this programmer for the popular erasable read only memories with part numbers 2704, 2708 and 2716. The product will program a 1 K chip in 2.5 minutes, a process which includes setting up the programmed pattern and verifying the pattern. The programmer is intended to be used with the user's processor as a peripheral, and comes in several models. The basic model is intended for use with 2704 (1/2 K bytes) and 2708 (1 K bytes) parts, borrowing power from the user's system. This model is priced at $199. The larger models feature built-in power supplies and manual operations via switches and LED readouts.■

Circle 486 on inquiry card.

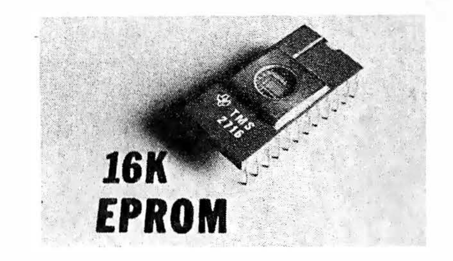
Nonvolatile to the Core

For the first time, to our knowledge, a product has been designed for the Altair (S-100) bus which provides *core memory* for a personal computer system. The product is Micro Memory Inc's MM-S100 8 K by 8 bit programmable memory card. Of what use is a magnetic core memory in an age of semiconductor circuits? Nonvolatility is the answer. With a core memory, magnetic storage of data is involved, a technology which is not dependent upon continuous application of power. Turn off the power on a core memory, and it will retain its pattern unaltered "forever." Turn on the power and the active circuits, and it is functionally like any semiconductor programmable memory. This core memory card thus combines the nonvolatility of a read only memory with the programmability of dynamic or static semiconductor memories.

The MM-S100 unit plugs directly into the Altair (S-100) bus, and has all the circuitry needed: timing, control logic, decode logic, drive circuits, address and data latches, power regulators, etc. It runs with a 1.0 μs cycle time so that no wait states are needed with a standard 8080 clock rate. The price is $650 from Micro Memory Inc, 9438 Irondale Av, Chatsworth CA 91311, (213) 998-0070.■

Circle 487 on inquiry card.

A 16 K Bit EROM



If your 2708 erasable programmable read only memories (EROMS) are filling up fast, here's one answer to the problem: Texas Instruments' 2716, a direct plug-in replacement for the 2708. Each chip contains 16,384 bits of memory and features low power consumption (375 milliwatts typical) and DC noise immunity in both high and low states so that all inputs can be driven by TTL logic without the use of pullup resistors.

The memory circuit is organized as 2048 words of 8 bit length. It is designed for high density, fixed memory applications where low power dissipation, fast turnarounds or program changes are required. Maximum access and minimum cycle times are 450 ns. The data outputs of the TMS2716 are three state to allow connecting of multiple devices on common bus. The EROM can be erased by exposing the chip through the transparent quartz lid to high intensity ultraviolet light. The TMS2716JL is supplied in a standard 24 pin dual in line ceramic package.

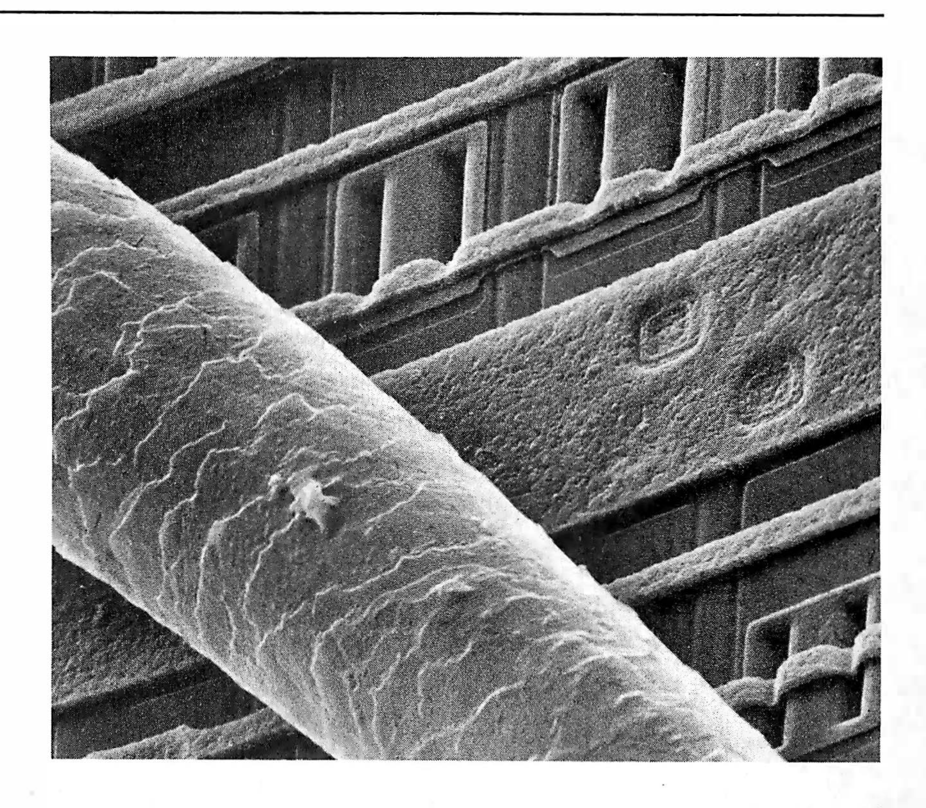
Contact Texas Instruments Inc, Inquiry Fulfillment Service, POB 1443, M/S 669 (attn: TMS2716) Houston TX 77001, (214) 238-2011.■

Circle 488 on inquiry card.

At the Frontiers of Silicon Technology

This electron microscope image shows a new American Microsystems Inc VMOS process memory device with a human hair juxtaposed on top of it. The magnification factor is on the order of 10,000 times the actual size. The V in VMOS is emphasized by the V-shaped slots in the structure of the devices. The part design from which this enlargement was made (the S4015-3 integrated circuit) is a new commercial volatile memory product which has an extremely fast access time (45 ns) and 1 K by 1 bit static operation. The product is intended for use with fast random access scratch pads, buffers, cache memories, etc. For those implementing microprogrammed machines on an experimental basis, this memory will prove ideal in a control store matched to the characteristics of the TTL bit slice parts such as the 2900 and the Texas Instruments' 74S481 family. American Microsystems Inc is located at 3800 Homestead Rd, Santa Clara CA 95051, (408) 246-0330.■

Circle 489 on inquiry card.



master charge

MICROCOMPUTER

your BANKAMERICARD welcome

8080A SUPPORT DEVICES

8212	4.00
8214	12.95
8216	5.25
8224	6.00
8228	9.25
8238	8.20
8251	12.00
8253	28.00
8255	12.00
8257	22.00
8259	22.00

6800 SUPPORT

6810P	6.00
6820P	8.00
6828P	9.60
6834P	21.95
6850P	12.00
6852P	17.00
6860P	15.00
6862P	18.00
6880P	2.70

Z80 SUPPORT DEVICES

3881	15.95
3882	15.95

F-8 SUPPORT DEVICES

3851	14.95
3852	14.95

FLOPPY DISC CONTROLLER

PD372D	65.00
1771	69.95

DYNAMIC RAMS

414D (16P)	5.50
1103 (16P)	1.50
2104 (16P)	6.50
2107B (22P)	4.50
2107B-4 (22P)	4.00
TMS4050 (18P)	4.50
TMS4060 (22P)	4.50
4096 (16P)	5.50
MM5262 (22P)	3.00
MM5270 (18P)	5.00
MM5280 (22P)	6.00

STATIC RAMS

31L01	2.00
91L11A	4.25
91L12A	4.25
1101A	1.00
2101	3.00
2102 (10S)	1.25
2102-1 (5.00NS)	1.50
2M1A-4	4.45
2112A-4	3.00
2501B	1.45
3107	2.95
*4200A (250NS)	13.75
410D (200NS)	11.95
*4804	20.00
5101	20.00
74C89	3.00
74S201	4.75
91L02A	2.00
7489	2.25
8225	1.50
8599	1.50
82S09	9.00

*Limited supply.

MISC OTHER COMPONENTS

NH0025CN	1.75
NH0026CN	3.00
N8T20	4.00
N826	3.25
N8T97	1.45
74367	1.00
DM8098	1.00
1488	1.95
1489	1.95
3205	6.20
D-3207A	2.50
C-3404	3.95
P-3408A	6.75
P-4201	4.95
MM-5320	7.50
MM-5369	2.00
DM-8130	3.00
DM-8131	2.50
DM-8831	2.50
DM-8833	2.50
DM-8835	2.50
SN74LS367	1.00
SN74LS368	1.00

MICROPROCESSOR'S

F-8	19.95
Z-80	36.95
Z-80A	49.95
CDP1802DC	29.50
AM2901	22.95
6502	24.95
6800	24.95
8008-1	8.75
8080A	15.95
8080B	16.95

SHIFT REGISTERS DYNAMIC

1404AN	3.00
2405	4.95
2505K	3.00

SHIFT REGISTERS STATIC

MM506	.89
2509K	1.00
2518B	3.95
2533V	2.00
TMS3002	1.00
TMS3112	3.95
MM5058	2.00

FIFO

3341A	6.75
2812-D	11.95

KEYBOARD CHIPS

AY5-2376	14.95
AY5-3600	14.95

TV GAME CHIPS

TMS 1955 (6 Games)	10.95
AYSS-8500 (6 Games)	10.95

USRT

S-2350	13.50
IM-6403	10.80
TMS-6011 (TI)	6.25
TR-1602A (WD)	6.25

UARTS

AY5-1013	6.75
AY5-1014A	9.95

CHARACTER GENERATORS

2513	6.75
2513	6.75
3257	18.00
MCM6571	10.80
MCM6571A	10.80
MCM6572	10.80
MCM6581	8.75

WAVEFORM GENERATOR

8038	4.50
MC4024	2.75
566	2.00

PROM'S

1702A	5.00	5204AQ	10.00
1702AL	7.00	6834	21.95
2704	20.00	6834-1	16.95
2708	24.00	82S23B	4.00
2716	75.00	82S129B	4.25
3601	4.50	8223B	4.00
5203AQ	7.00		

IMSAI/ALTAIR **S-100** COMPATIBLE

JADE Z80 KIT

—with PROVISIONS for ONBOARD 2708 and POWER ON JUMP

$135.00 EA.

Assembled & Tested

8K STATIC RAM BOARD

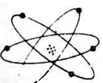
250ns.	$209.95
350ns.	$199.95
450ns.	$189.95

* WILL WORK WITH NO FRONT PANEL
* FULL DOCUMENTATION
* FULLY BUFFERED
* S100 DESIGN
* ADEQUATELY BYPASSED
* LOW POWER SCHOTTKY SUPPORT IC'S

KIT

250ns.	$169.95
350ns.	$149.95
450ns.	$139.95

JADE CO



Electronics for the Hobbyist and Experimenter

5351 WEST 144th STREET
LAWNDALE, CALIFORNIA 90260
(213) 679-3313

Discounts available at OEM quantities. Add $1.25 for shipping. California residents add 6% sales tax.

Circle 78 on inquiry card.

What's New?

PUBLICATIONS

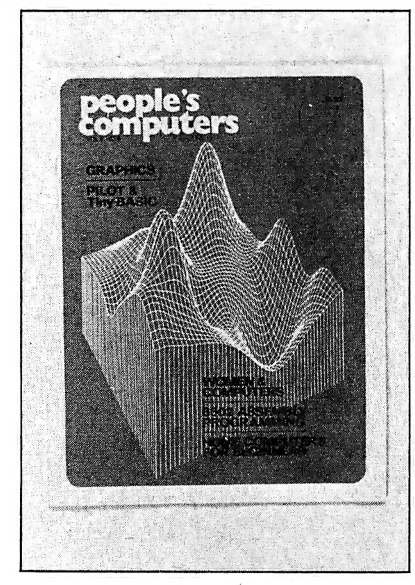
Motorola's New HEP Catalog



Motorola has announced its new cross reference guide and catalog describing the HEP line of semiconductor products. HEP products are designed primarily for hobbyists, experimenters, professional service technicians and dealers and consist of replacements for a large number of transistors, thyristors, diodes and FETs, as well as RTL, HTL, DTL, TTL and CMOS integrated circuits and linear devices. The catalog costs $2 and is available from the Motorola Technical Center, Motorola Semiconductor Products Inc, POB 20294, Phoenix AZ 85036, (602) 244-6900.■

Circle 456 on inquiry card.

Is the Dragon a Phoenix?



The theme of rebirth and renewal is a very real one, as exemplified by Phyllis Cole's transformation of *People's Computers* from a newspaper format tabloid (hard to keep track of) into the 64 page (including covers) saddle stitched publication shown in its May-June 1977 form in this photo. *People's Computers* is published bimonthly by People's Computer Company, 1263 El Camino Real, Box E, Menlo Park CA 94025. PCC is a tax-exempt, nonprofit corporation and donations are said to be tax-deductible. Subscriptions are $8 per year in the US. Single copy price is $1.50.

The editorial flavor which Phyllis brings to this publication is that of commentary on what's happening, light software, background information on computing and related peripheral issues. It is a magazine intended to be readable and enjoyable for the neophyte. (Our resident noncomputer people at BYTE grabbed the first issue so quickly that it became difficult to find a copy from which to abstract this short review.) Some titles from the first issue received here in the new format include:

- Home Computing: An Introduction for Novices
- Once Upon a Faire
- Computers and Copyright Law
- Women and Computers: A Dialogue
- The Dot and the Line
- Stock Market Simulations
- BASIC Mortgages
- Exagon
- Women and Math Projects: Lawrence Hall of Science
- Space Colony: Living In a Garden of Illusions
- Fortran Man
- More Tiny BASIC
- Make Believe Computers
- Pilot
- The Data Handlers Users Manual, Part 3.
- Announcements
- Letters

It is an interesting and positive transformation which should be sampled to be believed. . .CH■

Circle 458 on inquiry card.

IEEE Offers Microprocessor Talks on Cassette

A recording of three talks given at a tutorial "How to Use Microprocessors" held at Stanford University in the Spring of 1976 is available on standard magnetic tape cassettes for $5 from the IEEE. The talks are by Dr Robert Noyce, chairman of the board, Intel Corporation, Floyd Kvamme, vice president, National Semiconductor, and Dr Adam Osborne, president, Osborne Associates.

Topics covered include future developments in microprocessors, the business aspects of producing them, and which of the current microprocessors is most suited to particular hardware requirements. The tutorial was sponsored by the IEEE Computer Society, the Electron Devices and the Reliability Groups, Santa Clara (Silicon) Valley Section.

To obtain your copy of the cassette tape, send a check for $5 to the IEEE Section Office, 701 Welch Rd, Palo Alto CA 94304. Notes on the blackboard presentations and view graphs of the speakers will be included.■

Circle 457 on inquiry card.

A Special Free Offer from Radio Shack

Radio Shack is offering five free copies of their new *Archer Semiconductor Reference Handbook* to any interested organization.

The 128 page handbook, which normally sells for $1.95, lists over 36,000 replacement transistors, diodes and other devices, and includes a cross-reference guide, sections on the care and handling of transistors, soldering precautions, how to test transistors, and a glossary.

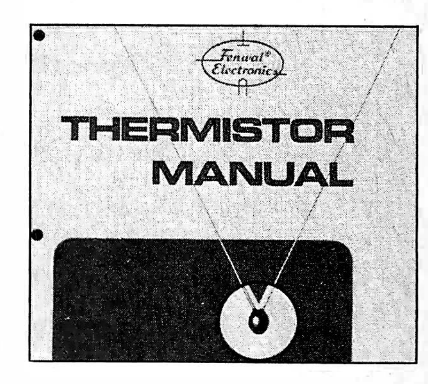
To get five free copies of the handbook, write on your club's stationery to Radio Shack, Dept SRH, 2617 W 7th St, Fort Worth TX 76107.■

Home Computer Books Available

Dilithium Press has a new brochure detailing their computer books, all of which are slanted toward the home computer hacker. Beginner's books as well as more advanced books are included in the list, available for free from Dilithium Press, POB 92, Forest Grove OR 97116.■

Circle 460 on inquiry card.

Fenwal Offers a New Thermistor Manual



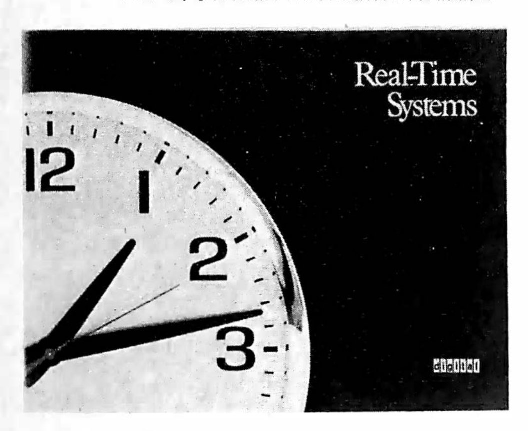
Many people who read BYTE are interested in microcomputer applications involving temperature measurement. One way to monitor temperatures is with a thermistor. Fenwal Electronics is making available a free 34 page thermistor manual containing a variety of temperature coefficient tables, resistance temperature tables and so on. Contact Fenwal Electronics, 63 Fountain St, Framingham MA 01701, (617) 872-8841.■

Circle 461 on inquiry card.

What's New?

SOFTWARE

PDP-11 Software Information Available



"Real-Time Systems," a new brochure available from Digital Equipment Corporation, describes the hardware and software components of Digital's real time computing systems based around the PDP-11 family of computers. The publication covers the RT-11, RSX and IAS operating systems, FORTRAN IV, FORTRAN IV-Plus and IAS COBOL high level languages. Also covered is special application oriented software for real time data acquisition, analysis and reporting in biological and physical science laboratories and process monitor/control situations. The brochure also lists sample configurations ranging from PDP-11V03 to PDP-11/70 systems, laboratory and industrial real time interfaces, and available supporting services. To obtain a copy, contact Communication Services, Digital Equipment Corporation, 444 Whitney St, Northboro MA 01532.■

Circle 500 on inquiry card.

New Information for Sphere Owners

If you've been looking for information about the Sphere computer, contact Programma Consultants, 3400 Wilshire Blvd, Los Angeles CA 90010. They offer a free catalog of new software and hardware, plus user group news pertaining to the Sphere. The catalog is arranged in question and answer format, and deals with such topics as the availability of FOCAL, FORTH, APL and cross compilers for the unit.■

Circle 501 on inquiry card.

An Assembler/Text Editor for the KIM

Micro Software Specialists have announced their new assembler/text editor package for KIM and TIM computers. Documentation and a hexadecimal object code listing are included. The price is $19.95 for the program in either cassette or paper tape form. Contact Micro Software Specialists, POB 3292, E T Station, Commerce TX 75428.■

Circle 502 on inquiry card.

A High Level Programming Language for the Motorola Microcomputer

Intermetrics Inc has announced PL/M6800, the first high level programming language for the Motorola M6800 (or AMI S6800) microcomputer. The language is syntactically identical to Intel's PLM.

PL/M6800 has a 1 pass compiler which produces directly loadable object code and listings. The new compiler features a user controlled switch to determine whether the emitted code will be in the AMI or Motorola loader format. Other user controlled features include listings of source code, object code, and assembler code, as well as symbol table dumps.

The new compiler is accessible via the NCSS timesharing network, or can be purchased directly from Intermetrics for installation on IBM 360 or 370 computers. The purchase price of $1000 includes a tape containing the cross compiler and all library routines, a user's manual, a language reference manual, and product maintenance for one year.

The PL/M6800 compiler is compatible with the PL/M language developed by Intel to program their line of microprocessors. Intermetrics claims to offer "true software portability" in that PL/M6800 is not only "PL/M-like," but is syntactically identical to PL/M.

Information on PL/M6800 is available from PL/M6800 Product Support, Intermetrics Inc, 701 Concord Av, Cambridge MA 02138, (617) 661-1840.■

Circle 503 on inquiry card.

A Mini Word Processing System

The Software Store has announced its Mini Word Processing system designed to run on Altair equipment under disk extended BASIC, for $150. Mini Word Processing is designed to help the operator generate letters, text, and mailing labels or envelopes. The system consists of seven programs which are driven by a menu select routine. Each program interacts with the operator to establish file names and drive numbers. The options are selected by the yes or no responses to the detailed program prompts. After each function is completed, the system reloads the menu routine.

A user's manual consisting of 51 pages is provided with the system. The manual includes detailed instructions concerning all operator prompts and system error messages, plus a number of examples with test data and programming considerations for custom applications.

Contact The Software Store at 706 Chippewa Sq, Marquette MI 49855, (906) 228-7622.■

Circle 504 on inquiry card.

Computerized Plotting

Sylvanhills Lab has announced the availability of 8080 software to control its series of plotters. Approximately 2 K bytes of memory are required. The software may be used in conjunction with application routines available from Micro-Visions Inc, 4926 Travis, Houston TX 77002.

Plotters are shipped completely assembled and tested. The user mounts them on the drawing surface and completes the interconnection between the control boards and the computer. An 8 bit parallel IO port, and 5 and 24 V power sources are also supplied by the user.

Applications include architectural, mechanical and schematic drawing; printed circuit board artwork; positioning of small objects; computer generated art; games. Sizes available are 11 by 17 inches (27.94 by 43.18 cm) for $750, 17 by 22 inches (43.18 by 55.88 cm) for $895, and 22 by 34 inches (55.88 by 86.36 cm) for $1200.

Contact Sylvanhills Lab Inc at 1 Sylvanway, POB 239, Strafford MO 65757, (417) 736-2664.■

Circle 505 on inquiry card.

TEMPOS, a Multitasking Operating System for MITS Computers

Administrative Systems Inc (ASI) has announced its memory resident, multiuser, multitasking operating system, the TEMPOS Operating System for MITS computers with MITS floppy disks. Up to seven on line users may access the system concurrently, using shared (reentrant) or different tasks. In addition, background tasks are supported as queued processes.

The TEMPOS system supports shared access to data files with a file "lock" feature under program control. Extensive file handling capabilities, including user defined logical record length and random access to file, as well as logical record number, are featured.

A command macro feature may be invoked under the TEMPOS system, allowing an unlimited number of macros to be defined and recalled at the system and user program levels. Also, to facilitate debugging, a single step trace feature is included for assembly language programs.

The minimum recommended memory requirement for the TEMPOS multiuser, multitasking operating system, using two disks and three terminals, is 48 K bytes. The price of the TEMPOS system is $1000. For further information contact Administrative Systems Inc, 222 Milwaukee, Suite 102, Denver CO 80206, (303) 321-2473.■

Circle 506 on inquiry card.

CRYSTALS
THESE FREQUENCIES ONLY

Part #	Frequency	Case/Style	Price
CY1A	1.000 MHz	HC33/U	$5.95
CY2A	2.000 MHz	HC33/U	$5.95
CY2.01	2.010 MHz	HC33/U	$1.95
CY3A	4.000 MHz	HC18/U	$4.95
CY7A	5.000 MHz	HC18/U	$4.95
CY12A	10.000 MHz	HC18/U	$4.95
CY14A	14.31818 MHz	HC18U	$4.95
CY19A	18.000 MHz	HC18/U	$4.95
CY22A	20.000 MHz	HC18/U	$4.95
CY308	32.000 MHz	HC18/U	$4.95

XR-2206KB Kit $19.95 **XR-2206KA Kit $14.95**

EXAR

WAVEFORM GENERATORS

XR-205	$8.40
XR-2206CP	4.49
XR-2207CP	3.85

STEREO DECODERS

XR-1310CP	$3.20
XR-1310EP	3.20
XR-1800P	3.20
XR-2567	2.99

MISCELLANEOUS

XR-2211CP	$6.70
XR-4136	99
XR-1468	3.85
XR-1488	5.80
XR-1489	4.80
XR-2208	5.20

TIMERS

XR-555CP	$ .49
XR-320P	1.55
XR-556CP	1.85
XR-2556CP	3.20
XR-2240CP	3.25

PHASE LOCKED LOOPS

XR-210	5.20
XR-215	6.60
XR-567CP	1.95
XR-567CT	1.70

CONNECTORS
PRINTED CIRCUIT EDGE-CARD
.156 Spacing-Tin-Double Read-Out

Bifurcated Contacts — Fits .054 to .070 P.C. Cards

15/30	PINS (Solder Eyelet)	$1.95
18/36	PINS (Solder Eyelet)	$2.49
22/44	PINS (Solder Eyelet)	$2.95
50/100	PINS (Wire Wrap)	$6.95
50/100A (.100 Spacing)	PINS (Wire Wrap)	$6.95

25 PIN-D SUBMINATURE

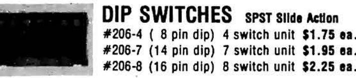
DB25P	PLUG	$3.25
DB25S	SOCKET	$4.95

HEAT SINKS



205-CB 680-.75A

205-CB	Beryllium Copper w-black finish for T0-5	$ .25
291-.36H	Aluminum for T0-220 Transistors & Regulators	$ .25
680-.75A	Black Anodized Aluminum for T0-3	$1.60
401-A	Black Anodized Aluminum — predrilled mounting holes for T0-3 — 4¾ x 1¼ x 2"	$1.75



DIP SWITCHES SPST Slide Action
- #206-4 (8 pin dip) 4 switch unit **$1.75 ea.**
- #206-7 (14 pin dip) 7 switch unit **$1.95 ea.**
- #206-8 (16 pin dip) 8 switch unit **$2.25 ea.**

Etching Kits (cannot be shipped via air)

32 X A-1	P.C. Etch Materials Kit enough for 5 circuit boards	$29.95 ea.
27 X A-1	Etched Circuit Kit Contains all — only add water	$ 9.95 ea.

Plugboards

3662	6.5 X 4.5 X 1/16 Epoxy glass P-Pattern-44 P.C. Tabs-spaced .156"	$ 6.95 ea.
22/44	Mating connector for plugboard — 22 pin double readout	$ 2.95 ea.
8800V	Universal Microcomputer/Processor plugboard — Epoxy Glass — complete with heatsink and mounting hardware 5.313 X 10 X 1/16 copper clad	$19.95 ea.

Vector

1/16 VECTOR BOARD

	0.1" Hole Spacing Part No.	P-Pattern L	W	Price 1	2-Up
PHENOLIC	64P44 062XXXP	4.50	6.50	1.72	1.54
	169P44 02XXXP	4.50	17.00	3.69	3.32
EPOXY GLASS	64P44 062	4.50	6.50	2.07	1.86
	84P44 062	4.50	8.50	2.56	2.31
	169P44 062	4.50	17.00	5.04	4.53
	169P84 062	8.50	17.00	9.23	8.26
EPOXY GLASS COPPER CLAD	169P44 062C1	4.50	17.00	6.80	6.12

Vector
SLIT-N-WRAP WIRE WRAP TOOL
- Slits and opens insulation exposing bare wire
- No pre-cutting or pre-stripping.
- Comes complete with two - 100 ft spools #28 AWG wire

Model P180 $24.50

HEXADECIMAL ENCODER 19-KEY PAD



- 1 - 0
- ABCDEF
- Return Key
- Optional Key (Period)
- — Key

$10.95 each

63 KEY KEYBOARD $24.95



This keyboard features 63 unencoded SPST keys, unattached to any kind of P.C.B. A very solid molded plastic 13" x 4" base suits most applications.

HD0165 16 LINE TO FOUR BIT PARALLEL KEYBOARD ENCODER CHIP **$7.95**

TOOLS

A97MS	— Diagonal Cutter - 4" semi-flush cut	$8.50 ea.
A11DMS	— Chain Nose Pliers - 4¾" long	7.50 ea.
T-6	— Wire Stripper - #16 to #26 gauge	3.75 ea.
55B	— Wire Stripper - #10 to #20 gauge	2.50 ea.
CS-8	— Cutter-Crimper Tool - 8¼" long	8.50 ea.
Nibbling Tool	— Cuts, Trims or Notches Metal up to #18 gauge	6.95 ea.
Nibbling Tool Replacement Punch		3.75 ea.

PERMACEL® P-29 PLUS Electrical Tape - All Weather
• ¾" wide x 66 ft. • Black vinyl
1-9 Rolls **$.79 each** 10-up Rolls **$6.95/10 roll package**

MICROPROCESSOR COMPONENTS

8080A	CPU	$16.00	8228	System Controller - Bus Driver	$10.95
8212	8 Bit Input/Output	4.95	MC6800	8 Bit MPU	22.50
8214	Priority Interrupt Control	15.95	MC6820	Periph. Interface Adapter	10.00
8216	Bi-Directional Bus Driver	6.95	MC6810AP1	128 x 8 Static RAM	7.95
8224	Clock Generator/Driver	10.95	MC6830L8	1024 x 8 Bit ROM	15.00
CDP1802		19.95	Z80	CPU	39.95

CPU'S

8080	Super 8008	24.95
8080A	Super 8008	16.00
2650	8 BIT MPU	26.50
P8085	CPU	29.95

SR'S

2504	1024 Dynamic	$ 3.95
2518	Hex 32 BIT	7.00
2519	Hex 40 BIT	4.00
2524	512 Dynamic	2.49
2525	1024 Dynamic	3.00
2527	Dual 256 BIT	3.95
2529	Dual 512 BIT	4.00
2532	Quad 80 BIT	3.95
2533	1024 Static	5.95
3341	Fifo	6.95
74LS670	16 x 4 Reg	3.95

UART'S

AY-5-1013	30K Baud	$5.95

ROM'S

2513(2140)	Char. Gen.-upper case	$ 9.95
2513(3021)	Char. Gen.-lower case	9.95
2516	Char. Gen.	10.95

RAM'S

1101	256 x 1	Static	$ 1.49
2101	256 x 4	Static	5.95
2102	1024 x 1	Static	1.75
2107/5280	4096 x 1	Dynamic	4.95
2111	256 x 4	Static	6.95
7489	16 x 4	Static	2.49
8101	256 x 4	Static	5.95
8111	256 x 4	Static	6.95
8599	16 x 4	Static	3.49
91L02	1024 x 1	Static	2.25
74200	256 x 1	Static	6.95
93421	256 x 1	Static	2.95
MM5262	2K x 1	Dynamic	2 for 1.00

PROMS

1702A	2048	Famos	$ 9.95
5203	2048	Famos	14.95
82S23	32 x 8	Open C	5.00
82S123	32 x 8	Tristate	5.00
74S287	1024	Static	7.95
3601	256 x 4	Fast	3.95
2708	8K	Eprom	19.95
2716	16K	Eprom	34.95
6301-1	1024	Tri-State Bipolar	3.49
6330-1	256	Open Collector Bipolar	2.95

SPECIAL REQUESTED ITEMS

FCM3817	$ 5.00	4N33	$ 3.95	5841	$ 9.95	9368	$ 3.95
AY-3-8500-1	8.95	8T20	7.50	MK50240	17.50	MC1408L7	9.95
MC3061P	3.50	8T97	2.00	11C90	19.95	LD110/LD111	25.00/set
MC4016P (74416)	7.50	9374	1.95	OS0026CH	3.75	AY-5-9100	17.50ea.
MCM6571	17.50	82S145	26.00	TIL308	10.50	95H90	11.95
MCM6574	17.50	MCM6575	17.50	ICM7208	22.00	ICM7209	7.50
AY-5-2376	14.95	ICM7045	24.95	ICM7207	7.50	HD0165	7.95

CLOCK CHIPS

MM5309	$9.95
MM5311	4.95
MM5312	4.95
MM5314	4.95
MM5316	6.95
MM5318	9.95
MM5369	2.95
CT7001	5.95

NOW IN STOCK

PARATRONICS
Featured on February's Front Cover of Popular Electronics

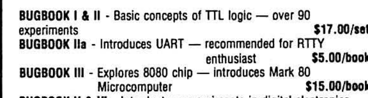
Logic Analyzer Kit
MODEL 100A $229.00/Kit

- Analyzes any type of digital system
- Checks data rates in excess of 8 million words per second
- Trouble shoot TTL, CMOS, DTL, RTL, Schottky and MOS families
- Displays 16 logic states up to 8 digits wide
- See ones and zeros displayed on your CRT, octal or hexadecimal format
- Tests circuits under actual operating conditions
- Easy to assemble — comes with step-by-step construction manual which includes 80 pages on logic analyzer operation.

• Some applications are:
— Troubleshooting microprocessor address, instruction, and data flow
— Examine contents of ROMS
— Tracing operation of control logic
— Checking counter and shift register operation
— Monitoring I/O sequences
— Verifying proper system operations during testing

PARATRONICS TRIGGER EXPANDER - Model 10
Adds 16 additional bits. Provides digital delay and qualification of input clock and 24-bit trigger word. — Connects direct to Model 100A for integrated unit). **Model 10 — $229.00** Baseplate — $6.95

BUGBOOK ®
Continuing Education Series



BUGBOOK I & II - Basic concepts of TTL logic — over 90 experiments **$17.00/set**
BUGBOOK IIa - Introduces UART — recommended for RTTY enthusiast **$5.00/book**
BUGBOOK III - Explores 8080 chip — introduces Mark 80 Microcomputer **$15.00/book**
BUGBOOK V & VI - Introductory experiments in digital electronics, 8080A Microcomputer interfacing and programming **$19.00/set**
555 TIMER APPLICATIONS SOURCEBOOK WITH EXPERIMENTS — over 100 design techniques **$6.95/book**
CMOS-M-DESIGNERS PRIMER AND HANDBOOK — a complete CMOS instruction manual **$6.00/book**
OP AMP MANUAL — over 25 experiments on all phases of OP AMPS **$9.00/book**
DBUG — a program for entering, debugging and storing assembly programs. **$5.00/book**
INSTRUCTORS MANUAL — Necessary for instruction on BUGBOOKS I and II **$3.00/book**

CONTINENTAL SPECIALTIES

PROTO BOARD 6 SPECIAL **$15.95** (6" long X 4" wide)

Other CS Proto Boards

PB100 - 4.5" x 6"	$ 17.95
PB101 - 5.8" x 4.5"	29.95
PB102 - 7" x 4.5"	39.95
PB103 - 9" x 6"	59.95
PB104 - 9.5" x 8"	79.95
PB203 - 9.75 x 6½ x 2¾	80.00
PB203A - 9.75 x 6½ x 2¾ (includes power supply)	129.95

PROTO CLIPS

14 PIN	$4.50
16 PIN	4.75
24 PIN	8.50

DESIGN MATES
- DM1 - Circuit Designer $69.95
- DM2 - Function Generator $74.95
- DM3 - RC Bridge $74.95

MAX-100 FREQUENCY COUNTER 20 MHz-100 MHz — 8 digit — .6" LED **$134.95**

Logic Monitor for DTL, HTL, TTL or CMOS Devices **$84.95**

QT PROTO STRIPS

QT type	#holes	price
QT-59S	590	12.50
QT-59B	bus strip	2.50
QT-47S	470	10.00
QT-47B	bus strip	2.25
QT-35S	350	8.50
QT-35B	bus strip	2.00
QT-18S	180	4.75
QT-12S	120	3.75
QT-8S	80	3.25
QT-7S	70	3.00

Experimentor 300	$ 9.95
Experimentor 600	$10.95

$5.00 Minimum Order — U.S. Funds Only
California Residents — Add 6% Sales Tax
Spec Sheets - 25¢ — Send 35¢ Stamp for 1978 Catalog
Dealer Information Available

NOW OPEN SATURDAYS

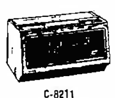
James ELECTRONICS

1021-A HOWARD AVE., SAN CARLOS, CA. 94070
PHONE ORDERS WELCOME — (415) 592-8097
All Advertised Prices Good Thru November

Timeband
A Trademark of Fairchild Camera and Instrument Corporation

Digital Alarm Clocks





- 24-hour alarm
- Doze Button
- 100% Solid State
- Large Red Led Display (.8" high)
- AM/PM Indicator
- Seconds display at touch of button
- Power failure indicator
- One year factory warranty

C-500 - Ivory Case
C-500B - Ebony Case
$16.95

C-8211 Woodgrain Case
$19.95

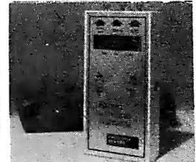
Timeband by FAIRCHILD



LAMP CLOCK
- 24 hour alarm
- Doze button
- Alarm-on indicator
- .8" high Red LED Display
- AM/PM Indicator
- High intensity lamp
- Lamp shuts off when collapsed

Model C-590 (Ivory) $29.95

ERC AM/FM 8-Track Stereo Receiver With BSR Changer



- PLL System
- BSR Record Changer
- Slide Controls
- Automatic AFC Control
- 4 Speaker Output
- Walnut finish vinyl covered wood veneer with smoke dust cover

Size: 20"W x 9⅜"H x 15¼"D
Model 8365 $149.95

DIGITAL STOPWATCH
- Bright 6 Digit LED Display
- Times to 59 minutes 59.59 seconds
- Crystal Controlled Time Base
- Three Stopwatches in One Times Single Event — Split & Taylor
- Size 4.5" x 2.15" x .90" (4½ ounces)
- Uses 3 Penlite Cells

Kit — $39.95
Assembled — $49.95
Heavy Duty Carry Case $5.95



Stop Watch Chip Only (7205) $19.95

imc 3½ DIGIT DPM KIT



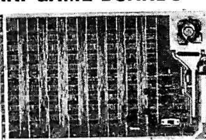
- New Bipolar Unit
- Auto Zeroing
- .5" LED
- Auto Polarity
- Low Power
- Single IC Unit

Model KB500 DPM Kit $49.00
Model 311D-5C-5V Power Kit $17.50

JE700 CLOCK
The JE700 is a low cost digital clock, but is a very high quality unit. The unit features a simulated walnut case with dimensions of 6" x 2½" x 1". It utilizes a MAN72 high brightness readout, and the MM5314 clock chip

12 or 24 Hour
115 VAC
KIT ONLY **$17.95**

Rejected ATARI GAME BOARDS



Over 60 Reusable IC's, misc. transistors, resistors, diodes, caps, crystals, switch, etc. (8½" x 16"). Limited Quantity.

$4.95 ea.

INSTRUMENT/CLOCK CASE
Injection molded unit. Complete with red bezal. 4½" x 4" x 1-9/16".

$3.95 ea.

JE803 PROBE



The LogicProbe is a unit which is for the most part indespensible in trouble shooting logic families TTL, DTL, RTL, CMOS. It derives the power it needs to operate directly off of the circuit under test drawing a scant 10 mA max. It uses a MAN3 readout to indicate any of the following states by these symbols (HI) - 1 (LOW) - 0 (PULSE) - P. The Probe can detect high frequency pulses to 45MHz. It can't be used at MOS levels or circuit damage will result

$9.95 Per Kit
printed circuit board

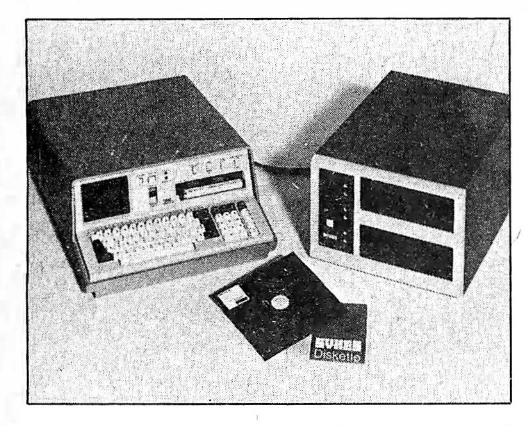
T²L 5V 1A Supply



This is a standard TTL power supply using the well known LM309K regulator IC to provide a solid 1 AMP of current at 5 volts. We try to make things easy for you by providing everything you need in one package, including the hardware for only **$9.95 Per Kit**

Circle 79 on inquiry card.

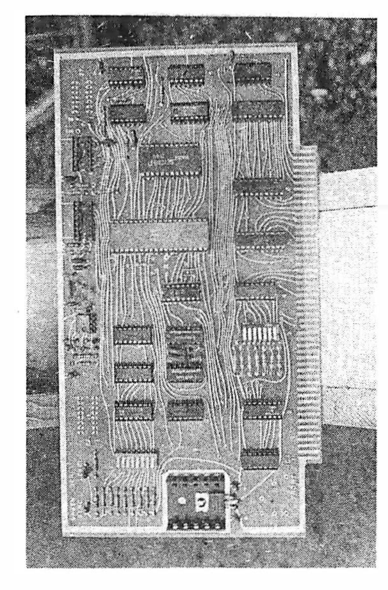
A 5100 System Mass Storage Device



Users of the IBM 5100 personal computer product will appreciate this new addition from an independent vendor. Sykes Datatronics of 375 Orchard St, Rochester NY 14606, (716) 458-8000, showed off this IBM 5100 compatible dual floppy disk subsystem at the NCC show in Dallas TX in June of this year. What it does is give the user a truly random access 3740 compatible diskette hardware subsystem and file management software on 3M cartridges for the 5100. No changes to the 5100 are required, and this subsystem plugs directly into the 5100's serial IO port. The software provided with this system includes ten BASIC files and 14 APL functions, and allows BASIC programs to communicate with APL programs using files on disk as an intermediary. The price is under $3000 for a single drive system, and under $4000 for dual drive.■

Circle 481 on inquiry card.

A New Cassette Recorder Interface

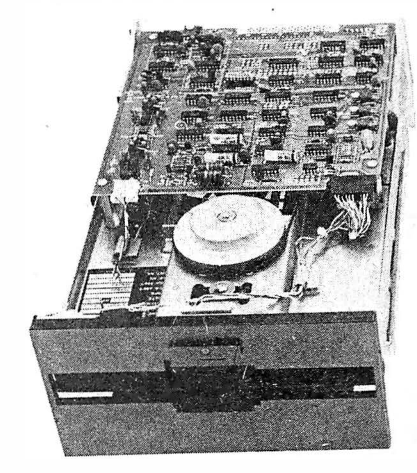


DaJen Electronics has announced a new cassette recorder interface with data transmission rates selectable from 800 to 12,000 bps. 12 K bytes of memory can be loaded in approximately 8 seconds. A 1 K byte monitor program is included to provide basic system operations and allow the saving of files. The unit is compatible with the Altair, iMSAI, Kansas City, Polymorphic Systems and Tarbell formats. Kit price is $120; the assembled unit costs $165. The price of the manual is $3.50.

Contact DaJen Electronics at 7214 Springleaf Ct, Citrus Heights CA 95610, (916) 723-1050.■

Circle 482 on inquiry card.

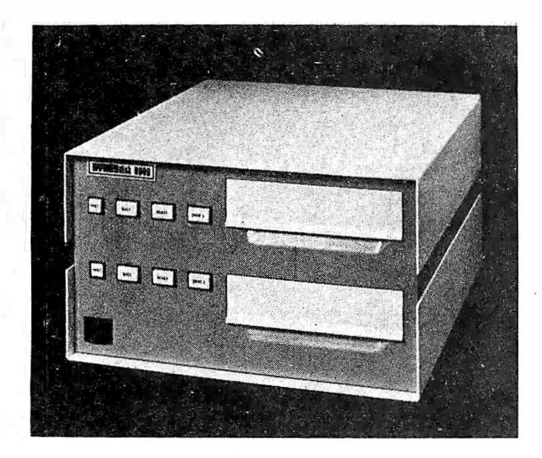
Floppy Disk Drive



According to the manufacturer, General Systems International Inc, 1440 Allec St, Anaheim, CA 92805, (213) 378-9385, this drive uses both sides of the floppy disk recording medium for data as opposed to just one. We can expect to see personal computing systems with on line floppy disk storage capacities on the order of 1.5 million bytes per drive growing out of this type of drive technology. Price of this drive to manufacturers is "in the low $400 range."■

Circle 484 on inquiry card.

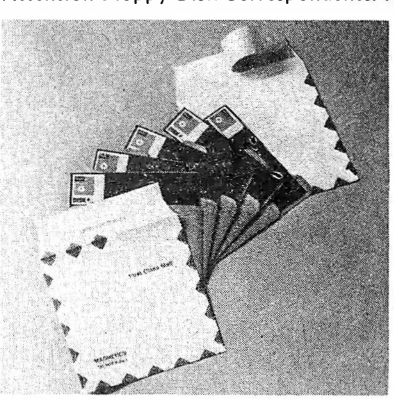
A Dual Floppy Subsystem



This photo shows the new Sykes Datatronics Series 9000 floppy disk system which is a complete mass storage subsystem with two drives and a built-in 6502 microprocessor controller. The system is offered in the dual drive version illustrated here ($3900) and a single drive version ($2800). All search, blocking, CRC verification and mechanical controls are handled asynchronously by the "smart" 6502-based controller of this device, and data is buffered using FIFO memories. The physical dimensions are table top compatible: 9.7 by 17 by 19 inches (25 by 43 by 48 cm). Hardware interfaces include an optional programmed IO parallel interface, which will be of interest to homebrewers, as well as detailed interfaces for a variety of microprocessors and minicomputers. Typical interface costs are $300 above the base prices; using the non-IBM format "dual and a half" density recording format, approximately 630,000 bytes can be recorded on each cartridge, making the dual drive on line capacity approximately 1.26 million bytes. Sykes is located at 375 Orchard St, Rochester NY 14606, (716) 458-8000.■

Circle 483 on inquiry card.

Attention Floppy Disk Correspondents...

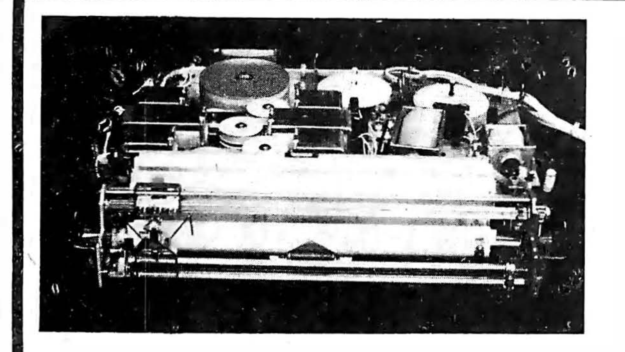


A new lightweight mailing envelope for floppy disks which saves 44 to 50 cents per disk in first class postage compared with older corrugated mailers is available from Curtis 1000.

The new envelope accommodates one to five floppy disks with filing sleeves in a lint and dust free environment.

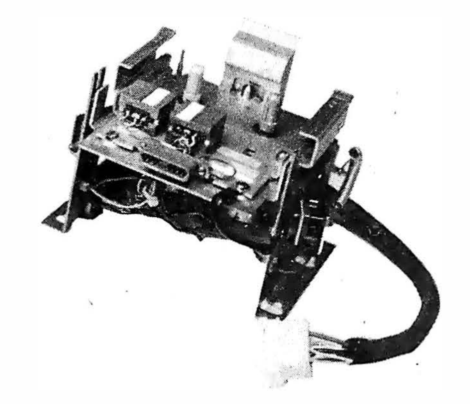
Made of DuPont's Tyvek® fiber, the new Curtis 1000 "Disk-O-Mailer" mailing envelope is extremely resistant to tearing and puncturing forces, as well as such dangers to floppy disks as chemicals and wetness. It features fast, dry sealing closure. Its glossy whiteness and green triangles printed along all edges on both sides assure first class handling in the post office just like regular business letters. Retailers and floppy disk software distribution outlets should contact the firm at 1000 Curtis Dr, Smyrna GA 30080, (404) 436-6155.■

Circle 485 on inquiry card.



MITE COMPUTER PRINTER

Mite 123P Impact printer. Designed for small keyboard printer terminals. 64 characters per line on 8½ inch paper. 75 characters per line, 10 CPS. Printer only, no electronics. With 30 pages documentation. Used, good shape.
Shipping wgt. 18 lbs. $63.00



VIATRON CASSETTE DECKS

The computer cassette deck alone $35.00

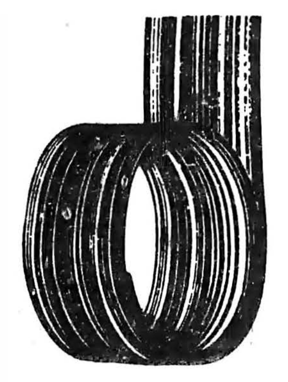


CONRAC VIDEO MONITOR

Used, checked out. Operates on 115 volts 60 cycle AC. In cabinets as shown. 128 x 40 with bandwidth of 8 Mc. Ideal for computer or TV monitor. Green phosphor display, 9" tube. With data & schematic.
Shipping wgt. 16 lbs. $62.00

SPECTRA FLAT TWIST

50 conductor, 28 gauge, 7 strands/conductor made by Spectra. Two conductors are paired & twisted and the flat ribbon made up of 25 pairs to give total of 50 conductor. May be peeled off in pairs if desired. Made twisted to cut down on "cross talk." Ideal for sandwiching PC boards allowing flexibility and working on both sides of the boards. Cost originally $13.00/ft
SP-324-A $1.00/ft. 10 ft/$9.00



SP-234-A $1.00 ft 50 cond. 10 ft/$9.00
SP-234-B .90 ft 32 cond. 10 ft/$8.00

WIRE WRAP WIRE

TEFZEL blue #30 Reg. price $13.28/100 ft. Our price 100 ft $2.00; 500 ft $7.50.

MULTI COLORED SPECTRA WIRE

Footage			10'	50'	100'
8 Cond.		#24	$2.50	9.00	15.00
12	"	22	3.00	11.00	18.00
14	"	22	3.50	13.00	21.00
24	"	#24	5.00	20.00	30.00
29	"	22	7.50	28.00	45.00

Great savings as these are about 1/4 book prices. All fresh & new.

TOUCHTONE ENCODER CHIP

Compatible with Bell system, no crystal required. Ideal for repeaters & w/specs. $6.00

CHARACTER GENERATOR CHIP

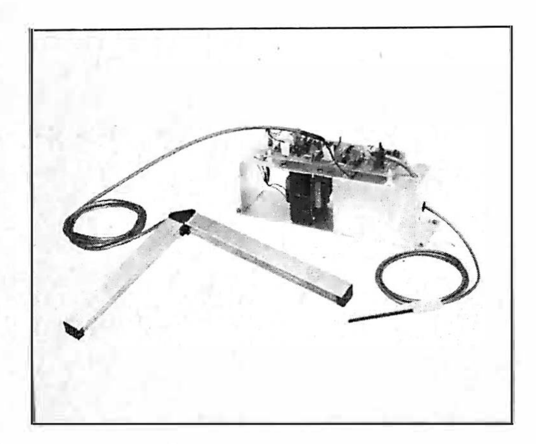
Memory is 512x5 produces 64 five by seven ASCII characters. New material w/data $6.00

Meshna

Please add shipping cost on above. Minimum order $10
FREE CATALOG NOW READY
P.O. Box 62, E. Lynn, Massachusetts 01904

Circle 83 on inquiry card.

Do You Want to Draw Pictures on Your Display?



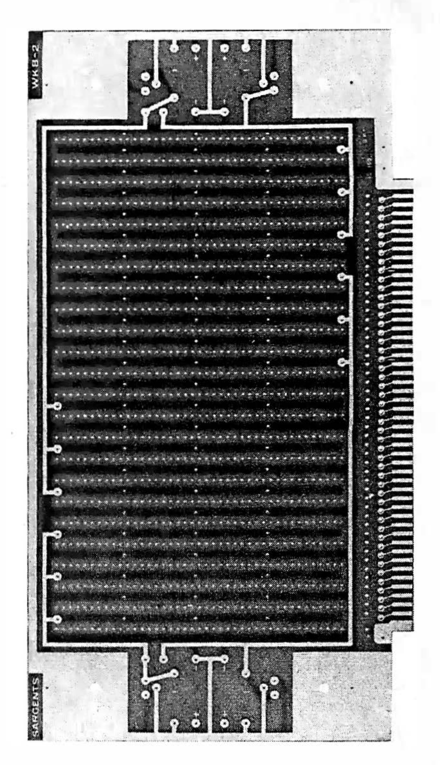
Users of personal computers who are interested in graphics input will find this new peripheral of great interest, and at a price which makes it attractive for personal use. Scientific Accessories Corporation, 970 Kings Highway W, Southport CT 06490, (203) 255-1526, has introduced the Model GP-101 sonic digitizer for a single unit price of $800. What you get is a stylus with or without ink or a cursor, an electronics package, and the L-frame sensor seen in this photo. It is intended to be used in any situation where input of XY position data is required. Typically, a computer oriented artist might make a rough sketch of graphic display information on paper and then trace the outline of the figure with the stylus after positioning the rough within range of the sensors. The artistic digitization could also be done interactively with the display in a freehand mode.

To use this device, some additional logic and timing circuitry will be required since the basic electronics simply produces TTL level signals which have a known start edge time and a variable delay time representing the X and Y distances to the stylus or cursor unit. The user must also provide power supplies and custom software to analyze the signals for particular purposes. ■

Circle 507 on inquiry card.

Sargent's New Altair (S-100) Prototype Board

Sargent's Distributing Company, 4209 Knoxville, Lakewood CA 90713, has introduced this new Altair (S-100) bus prototype board. The board is constructed of epoxy G-10 material. There is space for four 7805 type voltage regulators. It accommodates 14, 16, 18 24 and 40 pin wire wrap sockets with room for a maximum of forty-eight 14 or 16 pin sockets. Also available is a complete set of plans for a S-100 bus compatible front panel and bootstrap system which it features direct parallel ASCII keyboard input, one additional parallel input port, and two parallel output ports. This design uses PROMs for instant turn on and reset. Start cassette tape and your system is fully loaded and running in about 30 seconds, and can be wire wrapped using the prototyping board. Price of the prototype card is only $25 postpaid; the plan set is $7.50 postpaid; and a complete kit of all parts for the front panel design is $79.95 postpaid. ■



Circle 509 on inquiry card.

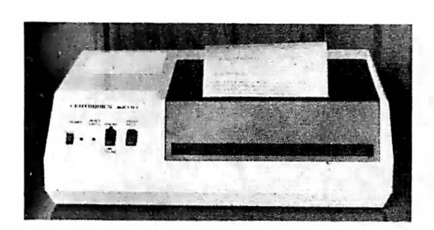
A New Model Self-Scan from Burroughs



The Electronic Components Division of Burroughs Corporation, POB 1226, Plainfield NJ 07061, has introduced this new single line 40 character version of Self-Scan II technology, intended for use in any product where a limited size alphanumeric display is required. Special effects include left or right data entry, moving message effects, blinking subfields within the 40 character line, etc. Self-Scan is a registered trademark of Burroughs Corporation. ■

Circle 510 on inquiry card.

Centronics Introduces High Speed Microprinter



Centronics Data Computer Corporation has announced a high speed compact microprinter called the Micro-1 for $595. Aimed at the home, hobby and microprocessor markets, the 240 character per second Micro-1 is offered as a complete unit including case, power supply, 96 character ASCII generator and interface, paper roll holder, low paper detector, bell, and multiline asynchronous input buffer.

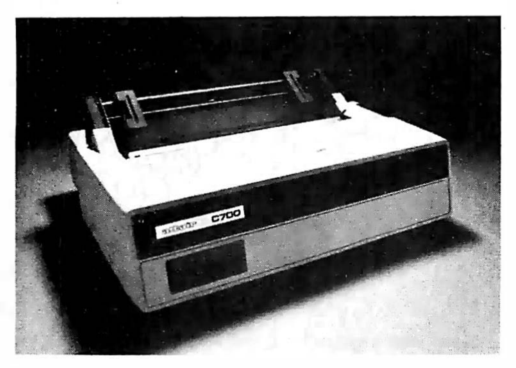
The microprinter produces copy on aluminum coated paper by discharging an electric arc to penetrate the coating, which is less than one micron thick. Toners and ribbons are not required.

The printed characters are said to be impervious to light, temperature and humidity. The machine prints 180 lines per minute on 4 3/4 inch roll paper in 20, 40 or 80 column widths, selectable by the user. The special aluminized paper used by the unit costs nominally more than standard paper.

This is an excellent way to get acceptably high speed listings at relatively low prices. Contact Centronics Data Computer Corporation, Hudson NH 03051, (603) 883-0111. ■

Circle 511 on inquiry card.

A 60 Character per Second Printer



The Altair C700 from MITS is a 60 character per second serial printer using a 5 by 7 dot matrix and the 64 character ASCII subset. The unit is designed to be interfaced to the Altair 8800 computer and features automatic motor control, paper runaway inhibitor and automatic line feed after carriage return. The printer is bidirectional and can print 26 132 column lines per minute. It can accommodate 15 inch wide forms.

Dimensions are 7 by 28 by 24.5 inches (17.8 by 45.7 by 62.2 cm). Contact MITS, 2450 Alamo SE, Albuquerque NM 87106. ■

Circle 508 on inquiry card.

Now low-cost memory stacks up in reliability!

Introducing a new generation of ECONORAM* dynamics with SynchroFresh™ reliability

Meet ECONORAM* III with SynchroFresh™, the 8Kx8 dynamic memory for S-100 bus computers that really works. And uses less than half the power of static designs. And costs just $149 for an assembled 8K.

Unlike previous attempts at building a low-cost dynamic memory, ECONORAM* III is entirely reliable ... because of SynchroFresh™, a new approach to memory refresh that is simple, elegant and totally effective.

SynchroFresh™ was invented by George Morrow, designer of the original ECONORAM*. Instead of arbitrarily interrupting your CPU to perform memory refresh cycles, Morrow designed SynchroFresh™ to weave refresh invisibly into the natural timing of the S-100 bus. SynchroFresh™ circuitry simply monitors your computer's machine states, utilizing all of the normal opportunities for memory refresh. It's that simple.

And simplicity means reliability and dramatically lower cost. That's why a SynchroFresh™ design was chosen for the first ECONORAM* dynamic, to follow in the footsteps of the largest-selling static memories for personal computers.

ECONORAM* III with SynchroFresh™ is an 8Kx8 dynamic board, configured as two individually addressable 4K blocks for flexibility. It is available assembled, tested and warranteed for one full year for just $149. This unprecedented warrantee offers a full refund of purchase price if ECONORAM* III does not run reliably with your S-100 CPU—evidence of our confidence in its performance.

It is also available as a kit with complete assembly instructions and documentation for $159.

ECONORAM* III with SynchroFresh™, in assembled or kit form, may be ordered directly from Thinker-Toys™. Write 1201 10th Street, Berkeley CA 94710 or call (415) 527-7548. Call BAC/MC orders toll-free to 800-648-5311. Or ask your computer store to order it for you.

NEW LOW PRICE

$149

8K assembled, tested, warranteed 1 year



*ECONORAM is a trademark of Godbout Electronics.

Circle 97 on inquiry card.

PERIPHERALS

Apple II Features Built-in Color Capability



The Apple II is Apple Computer Inc's entry in the home computer market. The unit uses the MOS Technology 6502 processor and can display alphanumeric characters and video graphics in 15 colors using any standard color television set.

A BASIC language package is permanently stored in 6 K bytes of read only memory; execution speed is fast enough to run many video games. The integer BASIC language includes special functions related to color video display programming.

In both the color graphics mode and in the high resolution graphics mode, four lines of text may be optionally displayed at the bottom of the screen to annotate displays. The Apple II also features a built-in cassette interface.

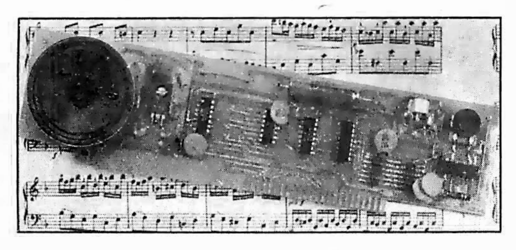
Minimum memory configuration available includes 4 K bytes of programmable memory and 8 K bytes of read only memory. A 2 K byte monitor provides debug commands, a mini-assembler, disassembler, floating point package and software-simulated 16 bit arithmetic capability.

The unit comes complete with a switching power supply which requires no fan. The computer is housed in a plastic case with dimensions of 18 by 15.25 by 4.5 inches (45.72 by 38.74 by 11.42 cm). It comes with two game paddles and a demonstration cassette for $1298. It is also available in board only form, without case, keyboard, power supply or accessories for $598.

Contact Apple Computer Inc, 20863 Stevens Creek Blvd, Bldg B3-C, Cupertino CA 95014.■

Circle 512 on inquiry card.

An Altair Bus Compatible Music Board

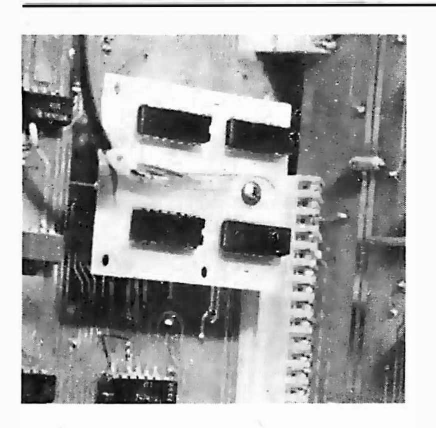


Newtech Computer Systems' Model 6 Music Board is designed to enable experimenters having Altair (S-100) bus computers to produce music and sound effects. Applications include generating melodies, rhythms, sound effects, Morse code and touch tone synthesis.

The Music Board comes assembled and tested. Features include selectable output port address decoding, a latched 6 bit digital to analog converter, audio amplifier, speaker, volume control and RCA phono jack for connection to external audio systems. It employs a glass epoxy printed circuit board with plated through holes and gold plated fingers.

A users manual, which is supplied with the board, includes a BASIC language program for writing musical scores and an 8080 assembly language routine for playing them. The price is $59.95. Contact Newtech Computer Systems Inc, 131 Joralemon St, Brooklyn NY 11201, (212) 625-6220.■

Circle 513 on inquiry card.



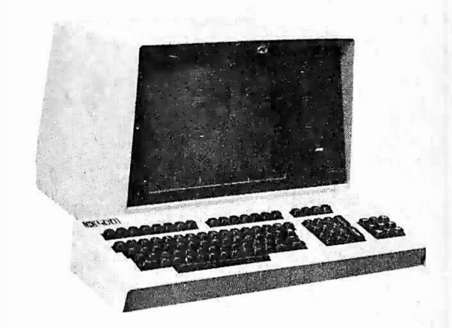
A CT-1024 Scroll Mod

Lenwood Computer Systems, POB 67, Hiawatha IA 52233, has announced a modification to Southwest Technical Products Corporation's CT-1024 terminal product. The Model SM-2 scrolling modification board is available at $19 plus $1.50 postage and handling, and converts the CT-1024 style display from a page oriented display to a scrolling display. The photo shows the board mounted on the CT-1024 main board using a stand off stud.■

Circle 514 on inquiry card.

A Pair of New Terminals





Infoton, Second Av, Burlington MA 01803, (617) 272-6660, has introduced this pair of video terminal products. The Model 200 is the low end version, a Teletype replacement with multiple additional features and 80 character by 24 line display. The Model 400 is a model with more features, including upper and lower case display, additional keyboard functions, etc. Both models feature RS-232, 20 mA current loop and 60 mA current loop serial interfaces at 16 switch selected data rates to 19,200 bits per second. No price was given in the documentation from which this note was abstracted.■

Circle 515 on inquiry card.

Several Gimix for the SwTPC 6800 Bus

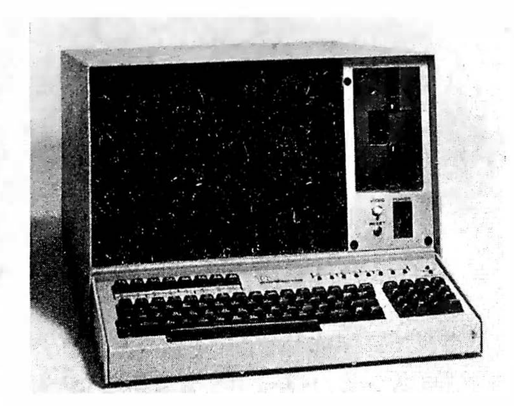
We received a sales brochure for three products available from Gimix Inc, 1337 W 37th Pl, Chicago IL 60609, (312) 927-5510, which plug directly into the Southwest Technical Products Corporation's 6800 bus. One of these products is a $119 read only memory board which holds up to 8 K of 2708 EROM parts (not supplied), and can be placed on any even 8 K memory address boundary (ie: 0000, 2000, 4000, 6000, 8000, A000, C000, or E000) using switches. A second product is a $25 extender board for use when troubleshooting or debugging a prototype card. The third product is a video output board which contains a 1 K by 8 bit volatile programmable memory region which can be connected at any 1 K boundary in memory address space using jumpers. This $249 generator can be set up for 16 lines of 32 characters or 16 lines of 64 characters; output is EIA video with adjustable "density"(?) and left-hand margin. These products are fully assembled.■

Circle 516 on inquiry card.

What's New?

PERIPHERALS

New Terminal from TEI



TEI Inc of Houston TX has announced a new processor terminal. Designated the Model MCS-PT, the unit is a self-contained computer system with display and disk storage, keyboard, and a 12 slot motherboard. It may be used either as a stand alone processor or as a processor terminal in a larger system.

Features include a 15 inch (38.1 cm) high resolution video monitor with a full upper and lower case ASCII character set keyboard, eight user designated special function keys, and a 16 key numeric cluster pad. A Shugart SA-400 minifloppy disk drive is standard.

The 12 slot mainframe contains a processor board featuring an 8080 processor and a special circuit that implements a start up "jump to" routine to any user selected memory address. 16 K bytes of programmable memory is provided with additional capacity available as an option. The disk controller (which can handle up to four drives) and the video board are also standard. The IO board provides three parallel and three serial ports with selectable data rates of 75 to 19,200 bps. RS-232C or TTL interfaces are provided. Power is provided by a constant voltage transformer (CVT) power supply.

Software provided includes a CP/M disk operating system and BASIC on disk. The processor terminal is $3495, fully assembled and tested. The kit is $2995. The unit without the disk drive and controller is $2495 assembled, or $2195 in kit form.

Contact Bill R Tatroe, CMC Marketing Corp, 7231 Fondren Rd, Houston TX 77036, or call (713) 774-9526.■

Circle 495 on inquiry card.

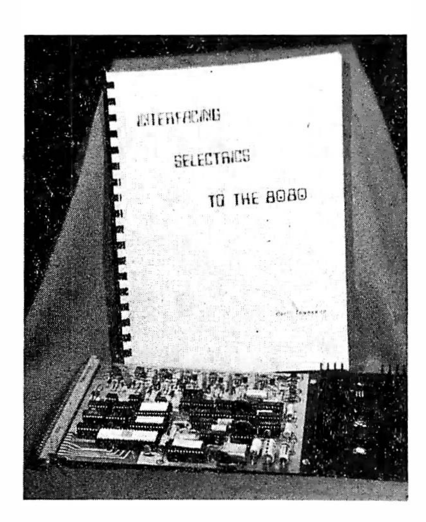
NEC "Spinwriter" Technology



The latest in a series of low inertia spinning plastic font impact printers to come to our attention is this NEC Information Systems "Spinwriter" terminal, intended for commercial markets. This terminal comes with a choice of five standard interfaces, 10 or 12 character per inch spacing (4 or 4.7 characters per cm), ASCII character codes, and a variety of plastic "thimble print mechanism" fonts for different type faces. The typing elements are rated at over three million impressions per character. No pricing information was given in the press release, other than the vague comment "10% below most competitive printers." If past experience is any guide this means a price above $3000 in unit quantities. Deliveries of this printer begin in October 1977 and are expected to be 60 days after receipt of orders thereafter. NEC Information Systems is located in Lexington MA.■

Circle 496 on inquiry card.

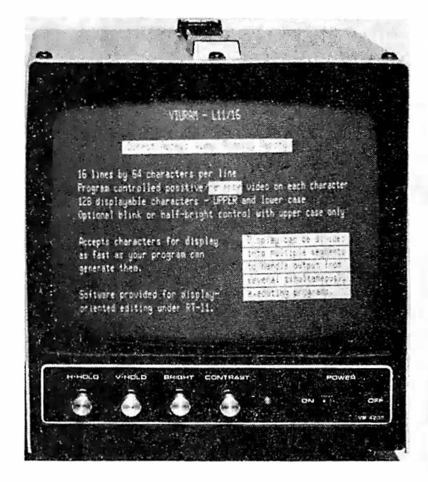
A Selectric Interface for Microcomputers



The Center for the Study of the Future has announced an electronic Selectric interface kit designed to work with most Selectric terminals and typewriters using the Tycom adapter. It includes 14 solenoid drivers rated for 24 V (solenoids and power supply are not included in the kit). The price of the kit plus manual is $325, available from the Center for the Study of the Future, 4110 NE Alameda, Portland OR 97212, (503)282-5835.■

Circle 497 on inquiry card.

Video Display Memory Board for LSI-11 Systems



Computer Technology, 6043 Lawton Av, Oakland CA 96418, (415) 451-7145, has introduced a board which plugs into the LSI-11 bus of Digital Equipment Corporation, and creates a video display peripheral which features 16 lines of 64 ASCII characters accessed as a 1 K byte region of memory address space. According to the information received here, the board fits into one dual width half slot segment of the LSI-11 backplane and requires only +5 V and +12 V power supplies. Output is EIA RS-170 composite video (2 volts peak to peak, negative sync) mA matched to 75 ohm coaxial cable. The photo shows a typical display on a video monitor. Individuals using LSI-11 systems may find this to be quite a useful addition to memory address space.■

Circle 498 on inquiry card.

A New Desktop Teleprinter Terminal for APL Users

Designed to meet the special character set requirements of APL users, the newly announced Anderson Jacobson AJ 860/A desktop Teleprinter terminal produces both an APL character set and a high resolution ASCII character set. Using a 9 wire dot matrix printer mechanism, the AJ 860/A prints each 9 by 5 character in a 9 by 12 character cell. The APL character set includes all of the standard APL overstrike characters, while the ASCII character set includes lower case, underscore and selectable double wide characters. Alternate selection of either the 128 code APL character set or the 128 code ASCII character set is done from the keyboard or remotely by code selection.

Standard features include operator selectable speeds of 10, 30, 45 or 60 cps, horizontal and vertical tabulation, reverse line feed, autopagination, dual gate forms tractor, self-test diagnostics, and a 350 character receive-only buffer with buffer overflow protection. Single unit purchase price is $3285 from Anderson Jacobson, 521 Charcot Av, San Jose CA 95131, (408)263-8520.■

Circle 499 on inquiry card.

NEW COMPUTER INTERFACE BOARD KIT

Our new computer kit allows you to interface serial TTL to RS 232 and RS 232 to TTL. There are four of these supplied with the kit, so you can run up to four devices on one TTL or four separate TTL to RS 232 devices.

Typical use: You can use your computer ports to run an RS 232 printer, video terminal and two other RS 232 devices at once, without constantly connecting and disconnecting your terminals.

$49.00

Example: Out store to printer — Voltage requirement +5V and ± 5V or ± 12 V depending on your RS 232 device.

We supply — board, connectors, documentation and components. Sorry, we do not supply case or power supply.

WHERE IT MAKES SENSE, MAY BE USED WITH ANY 8080, 6800, Z80 or F8 COMPUTER

GENERAL PURPOSE COMPUTER POWER SUPPLY KIT

This power supply kit features a high frequency torroid transformer with switching transistors in order to save space and weight. 115V 60 cycle primary. The outputs with local regulators are 5V to 10A, in one amp increments, – 5V at 1A, ± 12V at 1A regulators supplied 6 340T-5 supplied.

$79.00

UNIVERSAL 4K MEMORY BOARD KIT $69.95

This memory board may be used with the F8 and with minor modifications may be used with KIM-1µp.

32-2102-1 static RAM's, 16 address lines, 8 data lines in, 8 data lines out, all buffered. On-board decoding for any 4 of 64 pages, standard 44 pin, .156″ buss.

F8 EVALUATION BOARD KIT WITH EXPANSION CAPABILITIES

A fantastic bargain for only with the following features: **$99.00**

- 20 ma or RS 232 interface
- 64K addressing range
- Program control timers
- 1K of on-board static memory
- Built in clock generator
- 64 Byte register
- Built-in priority interrupts
- Documentation
- Uses Fairbug PSU

FOR FAIRBUG 4K F8 BASIC ON PAPER TAPE $25.00

2708-8K EPROM	$20.95
2522 STATIC SHIFT REG	$ 1.95
2513 CHARACTER GEN	$ 9.95
2518-HEX 32 BIT SR	$ 3.50
2102-1 1024 BT RAM	$ .99
5280 4K DYNAMIC RAM	$ 4.75
5202A UV PROM	$ 6.95
1101A-256 BIT RAM	$ .75
21078	$ 3.75
MK4008P	$ 1.95
1702A UV PROM	$ 4.95
5204.4K PROM	$10.95
82S23	$ 1.95
AY-5-1013 UART	$ 6.95

MINIATURE MULTI TURN TRIM POTS 100, 500, 1K,2K,5K,10K,25K,50K,100K, 200K,500K 1 Meg. $.75 each. . . . . 3/$2.00

MULTI-TURN TRIM POTS Similar to Bourns 3010 style 3/16″x5/8″x1 1/4″. 50, 100, 1K, 10K, 50K ohms $1.50 ea. . . 3/$4.00

LIGHT ACTIVATED SCR's TO-18, 200V 1A . . . . $ 1.10

TRANSISTOR SPECIALS

2N3585 NPN Si TO-66	$ .95
2N3772 NPN Si TO-3	$ 1.60
2N456A PNP GE	$ .75
2N4908 PNP Si TO-3	$ 1.00
2N6056 NPN Si TO-3 Darlington	$ 1.70
2N5086 PNP Si TO-92	4/$ 1.00
2N4898 PNP TO-66	$ .60
2N404 PNP GE TO-5	5/$ 1.00
2N3919 NPN Si TO-3 RF	$ 1.50
MPSA 13 NPN Si TO-92	3/$ 1.00
2N3767 NPN Si TO-66	$ .70
2N2222 NPN Si TO-18	5/$ 1.00
2N3055 NPN Si TO-3	$ .70
2N3904 NPN Si TO-92	5/$ 1.00
2N3906 PNP Si TO-92	5/$ 1.00
2N5296 NPN Si TO-220	$ .50
2N6109 PNP Si TO-220	$ .55
2N3638 PNP Si TO-5	5/$ 1.00
2N6517 NPN TO-92 Si	3/$ 1.00

C/MOS (DIODE CLAMPED)

74C10– .22	4012– .22	4023– .22	4042– .78
74C193–1.50	4013– .40	4024– .75	4046–2.25
4001– .22	4015– .95	4025– .22	4049– .40
4002– .22	4016– .40	4027– .40	4050– .40
4006–1.20	4017–1.05	4028– .88	4055–1.50
4007– .22	4018–1.00	4029–1.10	4066– .80
4009– .42	4019– .25	4030– .22	4071– .27
4010– .42	4020–1.05	4033–1.50	4076–1.05
4011– .22	4022– .95	4035–1.10	

IN 4148 (IN914) . . . . 15/$1.00

MCA-81 OPTICAL LIMIT SWITCH $1.50

LED READOUTS

FND 359 C.C. .4″	$ .50	DL704-.3″ C.C.	$ .95
FND 70 C.C. .4″	$ .55	MAN-7-.3″ C.A.	$ .95
FND 503 C.C. .5″	$ .85	NS 33-3 dig. array	$ .75
FND 510 C.A. .5″	$ .85	DL 747 C.A. .6″	$1.95

PRINTED CIRCUIT BOARD

4-1/2″x6-1/2″ SINGLE SIDED EPOXY BOARD 1/16″ thick, unetched $.60 ea. . . . . . 5/$2.60

7 WATT LD-65 LASER DIODE IR $8.95

2N 3820 P FET	$ .45
2N 5457 N FET	$ .45
2N2646	$ .45
ER 900 TRIGGER DIODES	4/ $1.00
2N 6028 PROG. UJT	$ .65

CCD202-100X 100 charge coupled device image sensor $145.00

MCM6571A 7 x 9 upper & lower case character generator $15.95

VERIPAX PC BOARD

This board is a 1/16″ single sided paper epoxy board, 4½″x6½″ DRILLED and ETCHED which will hold up to 21 single 14 pin IC's or 8, 16, or LSI DIP IC's with busses for power supply connector. $4.00

MV 5691 YELLOW-GREEN BIPOLAR LED	$ .90
FP 100 PHOTO TRANS	$ .50
RED, YELLOW, GREEN or AMBER LARGE LED's	6/$1.00
1L-5 (MCT-2)	$ .75
MOLEX PINS	100/$1.00 1000/$8.00
10 WATT ZENERS 3.9, 4.7, 5.6, 8.2, 12,15,18,22,100,150 or 200V	ea. $ .60
1 WATT ZENERS 4.7, 5.6, 10, 12, 15 18 or 22V	ea. $ .25
MC6860 MODEM CHIP	$9.95

Silicon Power Rectifiers

PRV	1A	3A	12A	50A	125A
100	.06	.14	.30	.80	3.70
200	.07	.20	.35	1.15	4.25
400	.09	.25	.50	1.40	6.50
600	.11	.30	.70	1.80	8.50
800	.15	.35	.90	2.30	10.50
1000	.20	.45	1.10	2.75	12.50

SILICON SOLAR CELLS

2¼″ diameter .4V at 500 ma. $4.00

REGULATORS

309K	$ .95	340K-5,12,15 or 24V	$ .85
723	$ .50	340T-5, 6, 8, 12 15,18 or 24V	$1.10
LM 376	$ .60	78 MG	$1.35
320K-5 or 15V	$1.40	79 MG	$1.35
320T-5, 15 or 24V	$ .85		

RS232 CONNECTORS: DB 25P male . . . . . $2.95; DB 25S female $3.50

TANTULUM CAPACITORS

.22UF 35V	5/$1.00	10UF 10V	$ .25
.47UF 35V	5/$1.00	22UF 25V	$ .40
.68UF 35V	5/$1.00	15UF 35V	3/$1.00
1UF 35V	5/$1.00	30UF 6V	5/$1.00
2.2UF 20V	5/$1.00	33UF 35V	$ .40
3.3UF 35V	4/$1.00	47UF 20V	$ .35
4.7UF 15V	5/$1.00	68 UF 15V	$ .50
6.8UF 35V	4/$1.00		

M7001 ALARM CLOCK CHIP $5.75

NATIONAL MOS DEVICES

MM1402– 1.75	MM5057– 2.25
MM1403– 1.75	MM5058– 2.75
MM1404– 1.75	MM5060– 2.75
MM5013– 2.50	MM5061– 2.50
MM5016– 2.50	MM5555– 4.75
MM5017– 2.70	MM5556– 4.75
MM5055– 2.25	MM5210– 1.95
MM5056– 2.25	MM5260– 1.75

TTL IC SERIES

7400– .14	7445– .55	74151– .60
7401– .14	7446– .65	74153– .60
7402– .14	7447– .65	74154– .95
7403– .14	7448– .65	74155– .70
7404– .18	7450– .15	74157– .58
7405– .18	7472– .29	74161– .85
7406– .25	7473– .29	74163– .80
7407– .25	7474– .29	74164– .95
7408– .18	7475– .45	74165– .95
7409– .17	7476– .30	74173– 1.20
7410– .14	7480– .35	74174– .95
7411– .20	7483– .62	74175– .82
7412– .20	7485– .87	74176– .75
7413– .39	7486– .30	74177– .75
7414– .63	7489– 1.85	74180 .65
7416– .25	7490– .42	74181– 1.90
7417– .25	7491– .58	74190– 1.00
7420– .14	7492– .43	74191– 1.00
7425– .25	7493– .45	74192– .83
7426– .22	7494– .70	74193– .83
7427– .25	7495– .65	74194– .85
7430– .14	7496– .65	74195– .52
7432– .25	74107– .28	74196– .86
7437– .21	74121– .33	74257– 1.25
7438– .21	74123– .65	74279– .87
7440– .14	74125– .40	75324– 1.75
7441– .70	74126– .40	75491– .50
7442– .40	74150– .90	75492– .50

MINIATURE DIP SWITCHES
CTS-206-4 Four SPST switches in one minidip package. $1.75
CTS-206-8 Eight SPST switches in a 16 pin DIP package. $2.20

5-8V SPST Miniature reed, relay., normally open, 330 Ohm coil resistance. $.75, 3/$2.00

ALCO MINIATURE TOGGLE SWITCHES
MTA 106 SPDT $1.20
MTA 206 DPDT $1.70

Full Wave Bridges

PRV	2A	6A	25A
100			1.30
200	.75	1.25	2.00
400	.95	1.50	3.00
600	1.20	1.75	4.00

SANKEN AUDIO POWER AMPS

Si 1010 G 10 WATTS	$ 7.95
Si 1020 G 20 WATTS	$15.95
Si 1050 G 50 WATTS	$27.95

WSU-30-Hand wire wrap tool used to wrap, unwrap & Strip #30 wire $5.30

#24, EIGHT CONDUCTOR SPECTRA FLAT CABLE 10'/$1.50 100'/$13.50

DIP SOCKETS

8 PIN – .22	24 PIN – .40
14 PIN – .25	28 PIN – .50
16 PIN – .28	40 PIN – .60
18 PIN – .30	

#30 WIRE WRAP WIRE SINGLE STRAND 100'/$1.40

74LS SERIES / LINEAR CIRCUITS

74LS SERIES		LINEAR CIRCUITS
	74LS125 – .65	
74LS00 – .26	74LS126 – .65	LM 101 – .75
74LS02 – .26	74LS132 – .80	LM 301/748 .31
74LS03 – .26	74LS136 – .39	LM307 – .30
74LS04 – .28	74LS138 – .72	LM 308 – .95
74LS05 – .28	74LS139 – .72	LM 311 – .95
74LS08 – .28	74LS151 – .98	LM 318 –1.35
74LS09 – .28	74LS153 – .93	LM 319 – .95
74LS10 – .26	74LS155 – 2.00	LM 324 –1.05
74LS11 – .26	74LS156 – .95	LM 339 –1.10
74LS13 – .50	74LS157 – .98	LM 358 –1.40
74LS15 – .28	74LS160 –1.02	LM 370 –1.15
74LS20 – .26	74LS161 –1.02	LM 377 –2.50
74LS21 – .26	74LS162 –1.02	LM 380 – .95
74LS22 – .26	74LS163 –1.02	LM 381 –1.25
74LS26 – .33	74LS168 –1.10	LM 382 –1.25
74LS27 – .33	74LS169 –1.10	LM 537 –2.50
74LS30 – .26	74LS170 –1.72	LM 553 –2.50
74LS32 – .33	74LS173 –1.39	LM 555 – .44
74LS37 – .37	74LS174 –1.05	LM556 – .85
74LS38 – .37	74LS175 –1.22	NE540L –2.75
74LS40 – .27	74LS190 –1.50	560 –2.00
74LS42 – .88	74LS191 –1.50	565 –1.10
74LS47 – .79	74LS192 –1.75	566 –1.50
74LS51 – .26	74LS193 –1.75	567 –1.50
74LS54 – .26	74LS195 –1.25	703 – .90
74LS90 – .95	74LS196 – .99	709 – .25
74LS73 – .40	74LS197 – .99	710 – .35
74LS74 – .40	74LS221 –1.25	741C or V – .31
74LS76 – .40	74LS257 –1.35	747 – .65
74LS90 – .89	74LS258 –1.38	LM 1310 –2.50
74LS92 – .85	74LS279 – .75	1456 – .95
74LS93 – .85	74LS365 – .66	1458 – .60
74LS109 – .43	74LS366 – .66	CA3046 – .75
74LS112 – .43	74LS367 – .66	3900 – .49
74LS113 – .43	74LS368 – .66	8038CC –3.90
74LS114 – .43	74LS390 –2.20	UA791P5 –1.95

TRIACS / SCR'S

PRV	TRIACS 1A	TRIACS 10A	TRIACS 25A	SCR'S 1.5A	SCR'S 6A	SCR'S 35A
100	.40	.70	1.30	.40	.50	1.20
200	.70	1.10	1.75	.60	.70	1.60
400	1.10	1.60	2.60	1.00	1.20	2.20
600	1.70	2.30	3.60		1.50	3.00

Terms: FOB Cambridge, Mass. Send Check or Money Order. Include Postage, Minimum Order $5.00, COD'S $20.00

Send 25¢ for our catalog featuring Transistors and Rectifiers 145 Hampshire St., Cambridge, Mass.



SOLID STATE SALES
P.O. BOX 74B
SOMERVILLE, MASS. 02143 TEL. (617) 547-4005

WE SHIP OVER 95% OF OUR ORDERS THE DAY WE RECEIVE THEM

Circle 134 on inquiry card.

Reader Service

BOMB
BYTE's Ongoing Monitor Box

August 1977 BOMB results appear on page 38.

To get further information on the products advertised in BYTE, fill out the reader service card with your name and address. Then circle the appropriate numbers for the advertisers you select from this list. Add a 9 cent stamp to the card, then drop it in the mail. Not only do you gain information, but our advertisers are encouraged to use the marketplace provided by BYTE. This helps us bring you a bigger BYTE.

**Reader Service inquiries not solicited. Correspond directly with company.*

BOMB: BYTE's Ongoing Monitor Box

Article No.	Couldn't Care Less					My Personal Rating — Liked Somewhat					Wow!
1	0	1	2	3	4	5	6	7	8	9	10
2	0	1	2	3	4	5	6	7	8	9	10
3	0	1	2	3	4	5	6	7	8	9	10
4	0	1	2	3	4	5	6	7	8	9	10
5	0	1	2	3	4	5	6	7	8	9	10
6	0	1	2	3	4	5	6	7	8	9	10
7	0	1	2	3	4	5	6	7	8	9	10
8	0	1	2	3	4	5	6	7	8	9	10
9	0	1	2	3	4	5	6	7	8	9	10
10	0	1	2	3	4	5	6	7	8	9	10
11	0	1	2	3	4	5	6	7	8	9	10
12	0	1	2	3	4	5	6	7	8	9	10
13	0	1	2	3	4	5	6	7	8	9	10
14	0	1	2	3	4	5	6	7	8	9	10
15	0	1	2	3	4	5	6	7	8	9	10
16	0	1	2	3	4	5	6	7	8	9	10
17	0	1	2	3	4	5	6	7	8	9	10
18	0	1	2	3	4	5	6	7	8	9	10
19	0	1	2	3	4	5	6	7	8	9	10
20	0	1	2	3	4	5	6	7	8	9	10
21	0	1	2	3	4	5	6	7	8	9	10
22	0	1	2	3	4	5	6	7	8	9	10
23	0	1	2	3	4	5	6	7	8	9	10
24	0	1	2	3	4	5	6	7	8	9	10
25	0	1	2	3	4	5	6	7	8	9	10

.... Comments ....

____

____

____

____

____

BYTE SUBSCRIPTIONS

For a subscription to BYTE, please complete this card.

Name ______________________

(Title) ______________________

(Company) ______________________

Address ______________________

City ______________________

State ______ Zip ______ Country ______

	USA		Canada Mexico
	□ $12	1 year	□ $17.50
	□ $22	2 years	□ $33.00
	□ $32	3 years	□ $48.50

$25 Europe (air freight)
$25 Elsewhere (surface mail) $________ enclosed

Please remit in US funds—Thank you.

□ Check enclosed (**Bonus:** you'll receive one EXTRA issue!)
□ BankAmericard □ Master Charge □ Bill me

Card No. ______________________

Expiration date ______________________

Four digits above name — Master Charge only ______________________

Signature ______________________

Please allow eight weeks for processing. *Thank you*

BYTE READER SERVICE

For fastest service transfer mailer label from wrapper to space provided below. Requests cannot be honored unless zip code is given. This card valid for 90 days only.

Name ______________________

(Title) ______________________

(Company) ______________________

Address ______________________

City ____________ **State** ______ **Zip** ______

NOTE—If label is missing or defaced fill out carefully—PLEASE PRINT—this is only way to get requested material to you.

1	31	61	91	121	151	181	211	241	271	301	331	361	391	421	451	481	511	541	571	601	631
2	32	62	92	122	152	182	212	242	272	302	332	362	392	422	452	482	512	542	572	602	632
3	33	63	93	123	153	183	213	243	273	303	333	363	393	423	453	483	513	543	573	603	633
4	34	64	94	124	154	184	214	244	274	304	334	364	394	424	454	484	514	544	574	604	634
5	35	65	95	125	155	185	215	245	275	305	335	365	395	425	455	485	515	545	575	605	635
6	36	66	96	126	156	186	216	246	276	306	336	366	396	426	456	486	516	546	576	606	636
7	37	67	97	127	157	187	217	247	277	307	337	367	397	427	457	487	517	547	577	607	637
8	38	68	98	128	158	188	218	248	278	308	338	368	398	428	458	488	518	548	578	608	638
9	39	69	99	129	159	189	219	249	279	309	339	369	399	429	459	489	519	549	579	609	639
10	40	70	100	130	160	190	220	250	280	310	340	370	400	430	460	490	520	550	580	610	640
11	41	71	101	131	161	191	221	251	281	311	341	371	401	431	461	491	521	551	581	611	641
12	42	72	102	132	162	192	222	252	282	312	342	372	402	432	462	492	522	552	582	612	642
13	43	73	103	133	163	193	223	253	283	313	343	373	403	433	463	493	523	553	583	613	643
14	44	74	104	134	164	194	224	254	284	314	344	374	404	434	464	494	524	554	584	614	644
15	45	75	105	135	165	195	225	255	285	315	345	375	405	435	465	495	525	555	585	615	645
16	46	76	106	136	166	196	226	256	286	316	346	376	406	436	466	496	526	556	586	616	646
17	47	77	107	137	167	197	227	257	287	317	347	377	407	437	467	497	527	557	587	617	647
18	48	78	108	138	168	198	228	258	288	318	348	378	408	438	468	498	528	558	588	618	648
19	49	79	109	139	169	199	229	259	289	319	349	379	409	439	469	499	529	559	589	619	649
20	50	80	110	140	170	200	230	260	290	320	350	380	410	440	470	500	530	560	590	620	650
21	51	81	111	141	171	201	231	261	291	321	351	381	411	441	471	501	531	561	591	621	651
22	52	82	112	142	172	202	232	262	292	322	352	382	412	442	472	502	532	562	592	622	652
23	53	83	113	143	173	203	233	263	293	323	353	383	413	443	473	503	533	563	593	623	653
24	54	84	114	144	174	204	234	264	294	324	354	384	414	444	474	504	534	564	594	624	654
25	55	85	115	145	175	205	235	265	295	325	355	385	415	445	475	505	535	565	595	625	655
26	56	86	116	146	176	206	236	266	296	326	356	386	416	446	476	506	536	566	596	626	656
27	57	87	117	147	177	207	237	267	297	327	357	387	417	447	477	507	537	567	597	627	657
28	58	88	118	148	178	208	238	268	298	328	358	388	418	448	478	508	538	568	598	628	658
29	59	89	119	149	179	209	239	269	299	329	359	389	419	449	479	509	539	569	599	629	659
30	60	90	120	150	180	210	240	270	300	330	360	390	420	450	480	510	540	570	600	630	670

MENTION BYTE WHEN WRITING TO ADVERTISERS

PLACE
STAMP
HERE



READER SERVICE
Publications Inc
PO Box 319
Dover NJ 07801

PLACE
STAMP
HERE

BYTE SUBSCRIPTIONS
PO Box 361
Arlington MA 02174

PLACE
STAMP
HERE

BOMB—
BYTE's Ongoing Monitor Box

BYTE Publications Inc
70 Main St
Peterborough NH 03458

The first complete, low-cost microcomputer system for business, home or education

Radio Shack TRS-80



Includes CPU, memory, keyboard, display, power supply, cassette recorder

The computer that helps a small business think big — and grow bigger. The TRS-80 can greatly reduce the time you spend on payroll, accounting, inventory control and other clerical tasks. So you have more time for clients or customers. You don't have to be an expert in programming or electronics, because the TRS-80 is *not* a kit — it's wired, tested, U.L. listed, ready to use. The Z80-based system comes with 4K read/write memory and Radio Shack Level-I BASIC® stored in read-only memory. Memory expandable to 62K bytes. With programming instructions and ready for an expanding selection of prepared programs on cassettes. Designed and built in the USA by Radio Shack. Just 599.95!



Clip and Mail Coupon Today!

Mail to: Radio Shack, Dept. TRS-80 C015
205 N.W. 7th St., Ft. Worth, TX 76101

Send me more data on the TRS-80 microcomputer

- **Description of applications, software and peripherals available through Radio Shack • Owners' newsletter**
- **Price list • List of stocking stores and dealers**

NAME ______________________ APT. NO. ____

ADDRESS ______________________________

CITY ______________ STATE ________ ZIP ________

SOLD ONLY WHERE YOU SEE THIS SIGN:

Radio Shack®

A TANDY COMPANY • FORT WORTH, TEXAS 76102
OVER 6000 LOCATIONS IN NINE COUNTRIES

Circle 120 on inquiry card.

Price may vary at individual stores and dealers